



TUGAS AKHIR – TI 184833

**PENJADWALAN MULTI PRODUK PADA LINI *FILLING*
PARALLEL DENGAN PERTIMBANGAN *SETUP TIME*,
RELEASE TIME DAN *MACHINE ELIGIBILITY RESTRICTION*
(STUDI KASUS: PT BAYER INDONESIA)**

MONICA PRATIWI
NRP 02411540000045

PEMBIMBING:
Nurhadi Siswanto, S.T., MSIE., Ph.D
NIP. 197005231996011001

KO-PEMBIMBING:
Stefanus Eko Wiratno, S.T., M.T.
NIP. 197103171998021001

DEPARTEMEN TEKNIK INDUSTRI
Fakultas Teknologi Industri
Institut Teknologi Sepuluh Nopember
Surabaya 2018



FINAL PROJECT – TI 184833

**MULTI PRODUCT SCHEDULING IN PARALLEL FILLING
LINE CONSIDERING SETUP TIME, RELEASE TIME AND
MACHINE ELIGIBILITY RESTRICTION
(CASE STUDY: PT BAYER INDONESIA)**

MONICA PRATIWI
NRP 02411540000045

SUPERVISOR:
Nurhadi Siswanto, S.T., MSIE., Ph.D
NIP. 197005231996011001

CO-SUPERVISOR:
Stefanus Eko Wiratno, S.T., M.T.
NIP. 197103171998021001

DEPARTMENT OF INDUSTRIAL ENGINEERING
Faculty of Industrial Technology
Institut Teknologi Sepuluh Nopember
Surabaya 2018

LEMBAR PENGESAHAN

PENJADWALAN MULTI PRODUK PADA LINI *FILLING* *PARALLEL* DENGAN PERTIMBANGAN *SETUP TIME*, *RELEASE TIME* DAN *MACHINE ELIGIBILITY RESTRICTION* (STUDI KASUS: PT BAYER INDONESIA)

TUGAS AKHIR

Diajukan untuk Memenuhi Salah Satu Syarat Memperoleh Gelar Sarjana Teknik
Program Studi S-1 Departemen Teknik Industri
Fakultas Teknologi Industri
Institut Teknologi Sepuluh Nopember
Surabaya, Indonesia

Oleh:

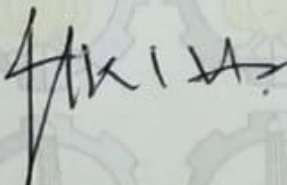
MONICA PRATIWI

NRP 02411540000045

Disetujui oleh:

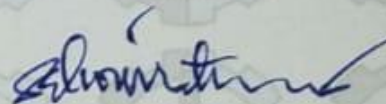
Pembimbing

Ko-pembimbing



Nurhadi Siswanto, S.T., MSIE., Ph.D

NIP. 197005231996011001



Stefanus Eko Wiratno, S.T., M.T.

NIP. 197103171998021001

SURABAYA, JANUARI 2019



**PENJADWALAN MULTI PRODUK PADA LINI *FILLING*
PARALLEL DENGAN PERTIMBANGAN *SETUP TIME*,
RELEASE TIME DAN *MACHINE ELIGIBILITY RESTRICTION*
(STUDI KASUS: PT BAYER INDONESIA)**

Nama	: Monica Pratiwi
NRP	: 0241154000045
Pembimbing	: Nurhadi Siswanto, S.T., MSIE., Ph.D
Ko-Pembimbing	: Stefanus Eko Wiratno, S.T., M.T.

ABSTRAK

PT Bayer Indonesia Divisi *Cropscience* memproduksi pestisida yang terdiri dari wujud *powder*, *liquid* dan pasta. Saat ini nilai OEE departemen produksi masih berada pada angka 68% yang disebabkan oleh rendahnya availabilitas. Availabilitas yang rendah disebabkan oleh besarnya *total duration loss* dengan faktor penunjang utama adalah *change over*. *Change over* terjadi apabila terdapat proses *setup*. Tingginya *change over* menandakan sering terjadi pergantian produk pada lini produksi. Penelitian ini bertujuan untuk menentukan penjadwalan produksi pada lini *filling parallel* sehingga dapat meminimasi terjadinya proses *setup*. Lini *filling parallel* terbagi menjadi dua jenis yaitu *identic parallel* dan *uniform parallel*. Dalam melakukan penjadwalan terdapat hal yang dipertimbangkan yaitu *sequence dependent setup time*, *release time* dan *machine eligibility restriction*. Permasalahan diselesaikan dengan pendekatan heuristik *Longest Processing Time* (LPT) – *Least Flexible Job* (LFJ), *Hungarian Method* dan kemudian hasil diperbaiki dengan *Variable Neighborhood Search* (VNS). Dalam penelitian ini juga dilakukan perbandingan model penjadwalan yang dimiliki oleh PT Bayer Indonesia Divisi *Cropscience* dengan algoritma yang dikembangkan. *Total duration loss* juga akan dibandingkan untuk melihat performansi dari algoritma yang dikembangkan. Dengan melakukan penerapan algoritma yang dikembangkan, *total duration loss* untuk produk *powder* berkurang sebesar 9.47% dan untuk produk *liquid* berkurang sebesar 17.56%.

Kata Kunci: *Least Flexible Job*, *Longest Processing Time*, Penjadwalan Produksi, Optimasi, *Variable Neighborhood Search*

(Halaman ini sengaja dikosongkan)

MULTI PRODUCT SCHEDULING IN PARALLEL FILLING LINE CONSIDERING SETUP TIME, RELEASE TIME AND MACHINE ELIGIBILITY RESTRICTION (CASE STUDY: PT BAYER INDONESIA)

Nama	: Monica Pratiwi
NRP	: 0241154000045
Pembimbing	: Nurhadi Siswanto, S.T., MSIE., Ph.D
Ko-Pembimbing	: Stefanus Eko Wiratno, S.T., M.T.

ABSTRACT

PT Bayer Indonesia Cropscience Division produces pesticide in form of powder, liquid and pasta. Current OEE of production department is still at 68% due to low availability value. Low availability value is caused by high total duration loss with changeover as the main supporting factor. Changeover occurs for each setup process on production line. High frequency of changeover indicates that frequent product change occurs production line. This research aims to develop production scheduling in parallel filling line that can minimize the frequency setup process. Parallel filling line is divided into two: identic parallel and uniform parallel. In developing production scheduling, there are considerations which are sequence dependent setup time, release time and machine eligibility restriction. This problem is solved with heuristic approach that divided into constructive phase and improvement phase. Constructive phase consists of Least Flexible Job (LFJ) – Longest Processing Time (LPT) and Least Flexible Job (LFJ) – Hungarian Method. Variable Neighborhood Search (VNS) is applied on the improvement phase. This research will compare the actual scheduling model to the proposed heuristic approach. Besides, this research will also compare total duration loss to see the performance of the proposed heuristic approach. With the implementation of proposed heuristic approach, total duration loss can be reduced up to 9.47% for scheduling powder product and up to 17.56% for scheduling liquid product.

Keyword: Least Flexible Job, Longest Processing Time, Production Scheduling, Optimization, Variable Neighborhood Search

(Halaman ini sengaja dikosongkan)

KATA PENGANTAR

Puji syukur kepada Tuhan yang Maha Esa karena kasihNya penulis dapat menyelesaikan tugas akhir yang berjudul “Penjadwalan Multi Produk pada Lini *Filling Parallel* dengan Mempertimbangkan *Setup Time*, *Release Time* dan *Machine Eligibility Restriction* pada PT Bayer Indonesia Divisi *Cropscience*” dengan baik. Tugas akhir ini merupakan tahap akhir yang harus diselesaikan penulis untuk mendapatkan Gelar Sarjana di Departemen Teknik Industri Institut Teknologi Sepuluh Nopember. Dalam pengerjaan tugas akhir ini penulis hendak mengucapkan terima kasih kepada pihak-pihak yang membantu hingga tugas akhir ini dapat diselesaikan dengan baik. Ucapkan terima kasih penulis berikan kepada:

1. Nurhadi Siswanto, S.T., MSIE., Ph.D, sebagai ketua departemen Teknik Industri sekaligus dosen pembimbing penulis dalam melakukan tugas akhir ini. Atas bimbingan, bantuan dan saran yang diberikan selama ini, penulis mengucapkan banyak terima kasih.
2. Stefanus Eko Wiratno, S.T., M.T., sebagai dosen ko-pembimbing penulis yang memberikan banyak wawasan dan masukan terhadap tugas akhir saya sehingga saya bisa menyelesaikan tugas akhir ini dengan baik.
3. Bapak Mario Christian, Bapak Awang Candra Wijaya dan Bapak Agung Sri Widodo sebagai pihak eksternal yang membantu penulis menyelesaikan tugas akhir dan bersedia memberikan waktunya untuk penulis. Khususnya dalam hal mendapatkan data di PT Bayer Indonesia serta memberikan izin bagi penulis untuk menggunakan PT Bayer Indonesia sebagai objek tugas akhir.
4. Prof. Dr. Budisantoso Wirjodirdjo, M. Eng dan Diesta Iva Maftuhah, S.T., M.T selaku dosen penguji seminar proposal dan sidang akhir yang telah banyak memberikan kritik dan saran terhadap tugas akhir penulis.
5. Ezra Simios Sucipto dan Maria Padmasih Faktarina sebagai kedua orang tua penulis yang senantiasa memberikan dukungan dan kasih yang tidak berhenti bagi penulis, juga kepada kakak dan adik penulis yang selalu memberikan penghiburan bagi penulis.

6. Mahasiswa teknik industri angkatan 2015 yang senantiasa membantu dan mendukung penulis selama masa perkuliahan.
7. Admin 2015 yang selalu membantu penulis menyelesaikan tugas akhir ini dan memberikan waktunya untuk berdiskusi mengenai topik yang diangkat penulis dalam tugas akhir. Serta Admin 2014 dan Admin 2016 yang selalu menjadi teman yang baik bagi penulis.
8. Pihak lain yang tidak dapat penulis sebutkan satu persatu tetapi sangat membantu penulis dalam menyelesaikan tugas akhir ini.

Dengan terselesaikannya tugas akhir ini, penulis berharap dapat memberikan manfaat bagi pihak-pihak yang ingin melakukan penelitian dalam hal penjadwalan operasi produksi dan khususnya dapat membantu PT Bayer Indonesia dalam mengurangi *total duration loss*.

Surabaya, Desember 2018

Penulis

DAFTAR ISI

ABSTRAK	i
ABSTRACT	iii
KATA PENGANTAR.....	v
DAFTAR ISI.....	vii
DAFTAR TABEL	xi
DAFTAR GAMBAR.....	xvi
BAB 1 PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	6
1.3 Tujuan.....	6
1.4 Manfaat.....	7
1.5 Batasan Penelitian	7
1.6 Asumsi Penelitian.....	7
1.7 Sistematika Penulisan.....	7
BAB 2 TINJAUAN PUSTAKA.....	11
2.1 <i>Manufacturing Planning and Control</i>	11
2.2 Pengertian Penjadwalan Operasi Produksi.....	13
2.3 Klasifikasi Penjadwalan Operasi Produksi	13
2.4 <i>Setup Time</i>	15
2.5 <i>Literature Review</i>	16
2.5.1 <i>Single Machine Group Scheduling with Family Setups to Minimize Total Tardiness</i>	16
2.5.2 <i>Parallel Machines Scheduling with Preference on Machines</i>	19
2.5.3 <i>Effective Heuristic Considering Machine Flexibility for Parallel Machine with Eligibility Problem</i>	21
2.5.4 <i>Minimizing Makespan on Parallel Machines with Release Time and Machine Eligibility Restriction</i>	22
2.5.5 <i>Single Machine Scheduling with Sequence Dependent Setup times and Delayed Precedence Constraint</i>	23
2.5.6 Penjadwalan Multi Produk pada Mesin Paralel <i>Filling Lithos</i>	25

2.6 Posisi Penelitian.....	29
BAB 3 METODOLOGI PENELITIAN	31
3.1 Tahap Pengumpulan Data.....	32
3.2 Tahap Pengembangan Model Penjadwalan Sistem Aktual	32
3.2.1 Penentuan Lini.....	32
3.2.2 Penentuan Urutan Produk.....	33
3.2.3 Perhitungan <i>Makespan</i>	35
3.3 Tahap Validasi Model Penjadwalan Sistem Aktual	35
3.4 Tahap Pengembangan Algoritma Fase <i>Constructive</i>	37
3.5 Tahap Pengembangan Algoritma Fase <i>Improvement</i>	38
3.6 Tahap Verifikasi dan Validasi Algoritma.....	38
3.7 Tahap Eksperimen	38
3.8 Tahap Komparasi <i>Total Duration Loss</i> dan <i>Penalty Overtime</i>	39
3.9 Tahap Analisis dan Interpretasi	39
3.10 Kesimpulan dan Saran	39
BAB 4 PENGUMPULAN DATA DAN PERANCANGAN ALGORITMA ..	41
4.1 Deskripsi Perusahaan.....	41
4.2 Pengumpulan Data.....	42
4.2.1 Alur Lini Produksi.....	42
4.2.2 Kapasitas Lini <i>Filling</i>	44
4.2.4 Matriks <i>Acceptance Contamination Limit (ACL)</i>	44
4.2.5 Jenis Produk dan Variasi Ukuran Produk	44
4.2.6 <i>Setup Time</i>	45
4.2.7 Data Permintaan Produk dan <i>Processing Time</i>	46
4.3 Perancangan Algoritma	46
4.3.1 Perancangan Algoritma Produk <i>Powder</i>	49
4.3.2 Perancangan Algoritma Produk <i>Liquid</i>	53
4.3.3 Perancangan Algoritma Produk MTO <i>Powder</i> dan <i>Liquid</i>	55
4.4 Verifikasi dan Validasi	56
4.4.1 Verifikasi dan Validasi Algoritma <i>Powder</i>	57
4.4.2 Verifikasi dan Validasi Algoritma <i>Liquid</i>	61
BAB 5 UJI COBA DAN ANALISIS	65

5.1 Implementasi Algoritma.....	65
5.1.1 Penjadwalan Produk <i>Powder</i> Per Bulan	65
5.1.2 Penjadwalan Produk <i>Liquid</i> Per Bulan	75
5.1.3 Rekap Hasil Penjadwalan <i>Powder</i> dan <i>Liquid</i>	85
5.2 Analisis Algoritma	86
5.2.1 Analisa Solusi Penjadwalan Produk <i>Powder</i> dan Produk <i>Liquid</i>	86
5.2.2 Analisa Perbandingan <i>Duration Loss</i> dan <i>Penalty Overtime</i>	87
5.2.3 Analisa Sensitivitas	88
BAB 6 KESIMPULAN DAN SARAN	95
6.1 Kesimpulan.....	95
6.2 Saran.....	96
DAFTAR PUSTAKA	97
LAMPIRAN.....	Error! Bookmark not defined.

(Halaman ini sengaja dikosongkan)

DAFTAR TABEL

Tabel 1. 1 Nilai OEE tahun 2015 – 2018	2
Tabel 2. 1 Posisi Penelitian	29
Tabel 3. 1 Permintaan Produksi Bulan Maret 2018	35
Tabel 3. 2 Opsi Lini tiap Kode Produk	36
Tabel 3. 3 Susunan Urutan Produk tiap Lini	37
Tabel 4. 1 Variasi Ukuran pada Lini <i>Filling</i>	Error! Bookmark not defined.
Tabel 4. 2 Kapasitas Maksimum Produksi 1 Shift	Error! Bookmark not defined.
Tabel 4. 3 Produk Tidak Boleh Berurutan	Error! Bookmark not defined.
Tabel 4. 4 Variasi Produk Wujud <i>Powder</i>	45
Tabel 4. 5 Variasi Produk Wujud <i>Liquid</i>	Error! Bookmark not defined.
Tabel 4. 6 Contoh Matriks <i>Setup Time</i> Produk <i>Powder</i>	45
Tabel 4. 7 <i>Demand</i> Produksi Produk <i>Powder</i>	Error! Bookmark not defined.
Tabel 4. 8 Contoh <i>Demand</i> Produksi Produk <i>Liquid</i>	46
Tabel 4. 9 Algoritma Tahap 1- <i>Generating Initial Solution</i> Produk <i>Powder</i>	51
Tabel 4. 10 Algoritma Tahap 2: <i>Improving Solution</i> - VNS	52
Tabel 4. 11 Algoritma Tahap 1: <i>Generating Initial Solution</i> Produk <i>Liquid</i>	54
Tabel 4. 12 Permintaan Produksi Produk MTS	57
Tabel 4. 13 Pengelompokan <i>Job</i> ke dalam Lini	57
Tabel 4. 14 Kemungkinan Urutan <i>Job</i> pada Lini 1	58
Tabel 4. 15 Kemungkinan Urutan <i>Job</i> pada Lini 2	58
Tabel 4. 16 Permintaan Produksi Produk MTO	61
Tabel 4. 17 Permintaan Produksi Produk <i>Liquid</i>	62
Tabel 4. 18 Solusi <i>Hungarian Method</i>	63
Tabel 5. 1 Nomor Kode Produk <i>Powder</i>	65
Tabel 5. 2 Matriks Produk-Mesin Bulan Maret dan April Produk <i>Powder</i>	66
Tabel 5. 3 Matriks Produk-Mesin Bulan September dan Oktober Produk <i>Powder</i>	66
Tabel 5. 4 Hasil LFJ Produk <i>Powder</i> Bulan Maret	68
Tabel 5. 5 Hasil Penjadwalan Produk <i>Powder</i> Bulan Maret	68

Tabel 5. 6 Urutan Produk <i>Powder</i> dalam Lini bulan Maret	68
Tabel 5. 7 Data Produk MTO Produk <i>Powder</i> bulan Maret.....	69
Tabel 5. 8 Hasil Penjadwalan Baru Produk <i>Powder</i> Bulan Maret	69
Tabel 5. 9 Urutan Baru Produk <i>Powder</i> dalam Lini bulan Maret	69
Tabel 5. 10 Hasil LFJ Produk <i>Powder</i> Bulan April	70
Tabel 5. 11 Hasil Penjadwalan Produk <i>Powder</i> Bulan April	70
Tabel 5. 12 Urutan Produk <i>Powder</i> dalam Lini bulan April	70
Tabel 5. 13 Data Produk MTO Produk <i>Powder</i> bulan April.....	70
Tabel 5. 14 Hasil Penjadwalan Baru Produk <i>Powder</i> Bulan April	71
Tabel 5. 15 Urutan Baru Produk <i>Powder</i> dalam Lini bulan April	71
Tabel 5. 16 Hasil LFJ Produk <i>Powder</i> Bulan September.....	71
Tabel 5. 17 Hasil Penjadwalan Produk <i>Powder</i> Bulan September	71
Tabel 5. 18 Urutan Produk <i>Powder</i> dalam Lini bulan September.....	72
Tabel 5. 19 Data Produk MTO Produk <i>Powder</i> bulan September	72
Tabel 5. 20 Hasil LFJ Produk <i>Powder</i> Bulan Oktober.....	73
Tabel 5. 21 Hasil Penjadwalan Produk <i>Powder</i> Bulan Oktober	73
Tabel 5. 22 Urutan Produk <i>Powder</i> dalam Lini bulan Oktober.....	73
Tabel 5. 23 Data Produk MTO Produk <i>Powder</i> bulan Oktober	74
Tabel 5. 24 Hasil Penjadwalan Baru Produk <i>Powder</i> Bulan Oktober.....	74
Tabel 5. 25 Urutan Baru Produk <i>Powder</i> dalam Lini bulan Oktober.....	74
Tabel 5. 26 Nomor Kode Produk <i>Liquid</i>	75
Tabel 5. 27 Nomor Kode Produk <i>Liquid</i> (Lanjutan)	75
Tabel 5. 28 Contoh Matriks Produk-Mesin Produk <i>Liquid</i>	76
Tabel 5. 29 Hasil LFJ Produk <i>Liquid</i> Bulan Maret	77
Tabel 5. 30 Hasil Hungarian Method bulan Maret.....	77
Tabel 5. 31 Hasil Penjadwalan Produk <i>Liquid</i> Bulan Maret	78
Tabel 5. 32 Urutan Produk <i>Liquid</i> dalam Lini bulan Maret	78
Tabel 5. 33 Data Produk MTO Produk <i>Liquid</i> bulan Maret.....	78
Tabel 5. 34 Hasil Penjadwalan Baru Produk <i>Liquid</i> Bulan Maret	79
Tabel 5. 35 Urutan Baru Produk <i>Liquid</i> dalam Lini bulan Maret	79
Tabel 5. 36 Hasil LFJ Produk <i>Liquid</i> Bulan April	79
Tabel 5. 37 Hasil <i>Hungarian Method</i> bulan April.....	79

Tabel 5. 38 Hasil Penjadwalan Produk <i>Liquid</i> Bulan April.....	80
Tabel 5. 39 Urutan Produk <i>Liquid</i> dalam Lini bulan April.....	80
Tabel 5. 40 Data Produk MTO Produk <i>Liquid</i> bulan April	80
Tabel 5. 41 Hasil Penjadwalan Baru Produk <i>Liquid</i> Bulan April.....	81
Tabel 5. 42 Urutan Baru Produk <i>Liquid</i> dalam Lini bulan April.....	81
Tabel 5. 43 Hasil LFJ Produk <i>Liquid</i> Bulan September	81
Tabel 5. 44 Hasil Hungarian Method bulan September.....	Error! Bookmark not defined.
Tabel 5. 45 Hasil Penjadwalan Produk <i>Liquid</i> Bulan September	81
Tabel 5. 46 Urutan Produk <i>Liquid</i> dalam Lini bulan September	82
Tabel 5. 47 Data Produk MTO Produk <i>Liquid</i> bulan September.....	82
Tabel 5. 48 Hasil Penjadwalan Baru Produk <i>Liquid</i> Bulan September	82
Tabel 5. 49 Urutan Baru Produk <i>Liquid</i> dalam Lini bulan September	83
Tabel 5. 50 Hasil LFJ Produk <i>Liquid</i> Bulan Oktober	83
Tabel 5. 51 Hasil Hungarian Method bulan Oktober.....	83
Tabel 5. 52 Hasil Penjadwalan Produk <i>Liquid</i> Bulan Oktober	84
Tabel 5. 53 Urutan Produk <i>Liquid</i> dalam Lini bulan Oktober	84
Tabel 5. 54 Data Produk MTO Produk <i>Liquid</i> bulan Oktober.....	84
Tabel 5. 55 Hasil Penjadwalan Baru Produk <i>Liquid</i> Bulan Oktober	85
Tabel 5. 56 Urutan Baru Produk <i>Liquid</i> dalam Lini bulan Oktober	85
Tabel 5. 57 Rekap Penjadwalan Produk <i>Powder</i> ...	Error! Bookmark not defined.
Tabel 5. 58 Rekap Penjadwalan Produk <i>Liquid</i>	Error! Bookmark not defined.
Tabel 5. 59 Perbandingan <i>Overtime</i>	Error! Bookmark not defined.
Tabel 5. 60 Hasil Penjadwalan Skenario Lini 1 Rusak.....	89
Tabel 5. 61 Urutan Produk Setelah Lini 1 Rusak...	Error! Bookmark not defined.
Tabel 5. 62 Pengaruh Permintaan Produksi terhadap Solusi	91
Tabel 5. 63 Perbandingan Sebelum dan Sesudah VNS Produk <i>Powder</i>	Error! Bookmark not defined.
Tabel 5. 64 Pengaruh Faktor Penurun Suhu dan Maks. Iterasi Produk <i>Powder</i>	Error! Bookmark not defined.
Tabel 5. 65 Perbandingan Sebelum dan Sesudah VNS Produk <i>Liquid</i>	Error! Bookmark not defined.

Tabel 5. 66 Pengaruh Faktor Penurun Suhu dan Maks. Iterasi Produk *Liquid*
..... **Error! Bookmark not defined.**

(Halaman ini sengaja dikosongkan)

DAFTAR GAMBAR

Gambar 1. 1 Data Permintaan Produksi 2018	1
Gambar 1. 2 Faktor Penyebab <i>Duration Loss</i>	3
Gambar 2. 1 Skema <i>Manufacturing Planning Control</i>	11
Gambar 2. 2 <i>Flow shop Process</i>	15
Gambar 2. 3 <i>Job Shop Process</i>	15
Gambar 2. 4 <i>Makespan Calculation Model</i>	24
Gambar 2. 5 <i>Job Series Move</i>	25
Gambar 2. 6 <i>Job Insertion</i>	25
Gambar 2. 7 <i>Job Exchange</i>	25
Gambar 3. 1 Diagram Alir Penelitian	31
Gambar 3. 2 <i>Flowchart</i> Penentuan Lini Produksi	33
Gambar 3. 3 <i>Flowchart</i> Penentuan Urutan Produk	34
Gambar 4. 1 <i>Flow Process</i> Produk Wujud <i>Powder</i>	42
Gambar 4. 2 Aliran Material Lini Produk Wujud . Error! Bookmark not defined.	
Gambar 4. 3 <i>Flow Process</i> Produk Wujud <i>Liquid</i>	43
Gambar 4. 4 Aliran Material Lini Produk Wujud <i>Liquid</i> Error! Bookmark not defined.	
Gambar 4. 5 <i>Influence Diagram</i> Permasalahan Penjadwalan	47
Gambar 4. 6 <i>Flowchart</i> Penjadwalan dengan Produk MTO	56
Gambar 4. 7 <i>Hungarian Method</i> dengan <i>Software Lingo</i>	62
Gambar 4. 8 Status Solusi <i>Hungarian Method</i>	63
Gambar 5. 1 <i>Ganttchart</i> Penjadwalan Bulan Maret Produk <i>Powder</i>	Error! Bookmark not defined.
Gambar 5. 2 <i>Ganttchart</i> Penjadwalan Bulan Maret dengan MTO Produk <i>Powder</i>	Error! Bookmark not defined.
Gambar 5. 3 <i>Ganttchart</i> Penjadwalan Bulan April Produk <i>Powder</i>	Error! Bookmark not defined.
Gambar 5. 4 <i>Ganttchart</i> Penjadwalan Bulan April dengan MTO Produk <i>Powder</i>	Error! Bookmark not defined.

Gambar 5. 5 *Ganttchart* Penjadwalan Bulan September Produk *Powder* **Error!**
Bookmark not defined.

Gambar 5. 6 *Ganttchart* Penjadwalan Baru Bulan September dengan MTO Produk *Powder* **Error! Bookmark not defined.**

Gambar 5. 7 *Ganttchart* Penjadwalan Bulan Oktober Produk *Powder* **Error!**
Bookmark not defined.

Gambar 5. 8 *Ganttchart* Penjadwalan Baru Bulan Oktober dengan MTO Produk *Powder* **Error! Bookmark not defined.**

Gambar 5. 9 *Ganttchart* Penjadwalan Bulan Maret Produk *Liquid*..... **Error!**
Bookmark not defined.

Gambar 5. 10 *Ganttchart* Penjadwalan Baru Bulan Maret dengan MTO Produk *Liquid* **Error! Bookmark not defined.**

Gambar 5. 11 *Ganttchart* Penjadwalan Bulan April Produk *Liquid*..... **Error!**
Bookmark not defined.

Gambar 5. 12 *Ganttchart* Penjadwalan Baru Bulan April dengan MTO Produk *Liquid* **Error! Bookmark not defined.**

Gambar 5. 13 *Ganttchart* Penjadwalan Bulan September Produk *Liquid* **Error!**
Bookmark not defined.

Gambar 5. 14 *Ganttchart* Penjadwalan Baru Bulan September dengan MTO Produk *Liquid* **Error! Bookmark not defined.**

Gambar 5. 15 *Ganttchart* Penjadwalan Bulan Oktober Produk *Liquid* **Error!**
Bookmark not defined.

Gambar 5. 16 *Ganttchart* Penjadwalan Baru Bulan Oktober Produk *Liquid* **Error!**
Bookmark not defined.

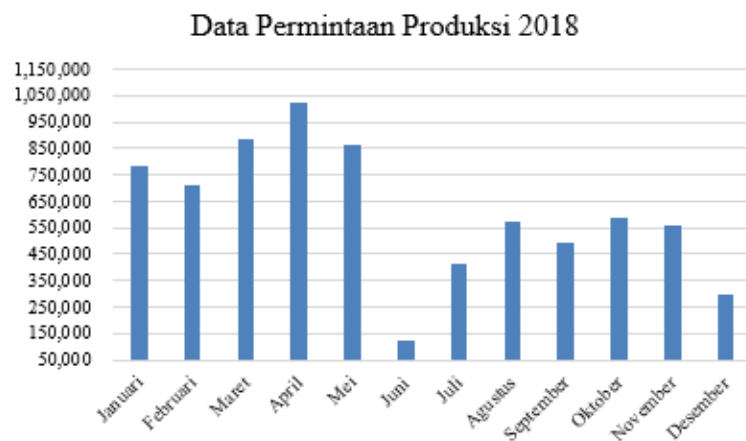
BAB 1

PENDAHULUAN

Bab ini akan dibahas terkait latar belakang masalah yang diangkat dalam penelitian kemudian akan dirangkum dalam rumusan permasalahan. Selain itu juga dijelaskan tujuan, manfaat dan ruang lingkup penelitian.

1.1 Latar Belakang

PT Bayer Indonesia terdiri atas tiga divisi yaitu *pharmaceuticals*, *health care* dan *cropscience* dimana Divisi *Cropscience* merupakan divisi yang secara khusus memproduksi produk yang berhubungan dengan pestisida. Pestisida yang diproduksi terdiri atas insektisida, fungisida dan herbisida dalam wujud *powder*, *liquid* dan pasta. Penelitian ini hanya berfokus pada produk *powder* dan *liquid* karena variasinya yang tergolong tinggi. Kemasan yang digunakan untuk produk *powder* beragam mulai dari ukuran 50 gram hingga ukuran 25 kilogram sedangkan untuk produk *liquid* mulai dari ukuran 50 ml hingga 200000 ml. Setiap tahunnya PT Bayer Indonesia Divisi *Cropscience* mampu memproduksi hingga lebih dari 7000 metric ton dengan grafik permintaan produksi pada tahun 2018 sebagai berikut:



Gambar 1. 1 Data Permintaan Produksi 2018

Berdasarkan Gambar 1.1 di atas dapat dilihat bahwa permintaan produksi tertinggi terjadi pada bulan April diikuti dengan bulan Maret dan Mei. Permintaan

produksi yang tinggi ini tentunya memberikan tantangan dalam proses pemenuhannya. Keadaan permintaan produksi yang tinggi terjadi pada lima bulan pertama sedangkan pada bulan berikutnya permintaan produksi cenderung pada keadaan normal dengan permintaan terendah pada bulan Juni. Proses produksi dilakukan untuk memenuhi permintaan produksi produk yang bersifat *Make to Stock* (MTS) atau *dependent demand* dan untuk produk yang bersifat *Make to Order* (MTO) atau *independent demand*. Jumlah dan variasi produk MTS telah ditentukan berdasarkan *Master Production Scheduling* (MPS) yang telah melewati tahap *Rough Cut Capacity Planning*. Sedangkan produk MTO tidak diketahui datangnya dan akan menjadi prioritas utama untuk diproduksi ketika produk tersebut datang.

Dalam memenuhi permintaan produksi tersebut, departemen produksi sebagai kunci utama perlu selalu ditingkatkan performansinya. Salah satu kriteria yang digunakan untuk mengukur performansi tersebut adalah OEE (*Overall Equipment Effectiveness*). OEE adalah *performance metrics* yang menunjukkan keefektifan dari suatu operasi produksi yang dilakukan oleh suatu perusahaan. OEE terdiri dari tiga hal utama yaitu *performance efficiency*, *availability* dan *quality rate*. Perhitungan OEE di PT Bayer Indonesia Divisi *Cropscience* telah terintegrasi dalam *database* internal yang berguna untuk menghitung dan menyimpan OEE.

Standar OEE industri proses kelas dunia berada pada nilai 85% (Chandran, 2009). Melihat tingginya nilai OEE tersebut sebagai pembandingan, setiap tahunnya PT Bayer Divisi *Cropscience* terus berupaya meningkatkan nilai OEE-nya sebagai salah satu bentuk *improvement* produksi. Data OEE gabungan dari seluruh lini produksi yang dimiliki oleh PT Bayer Indonesia Divisi *Cropscience* ditampilkan pada Tabel 1.1.

Tabel 1. 1 Nilai OEE tahun 2015 – 2018

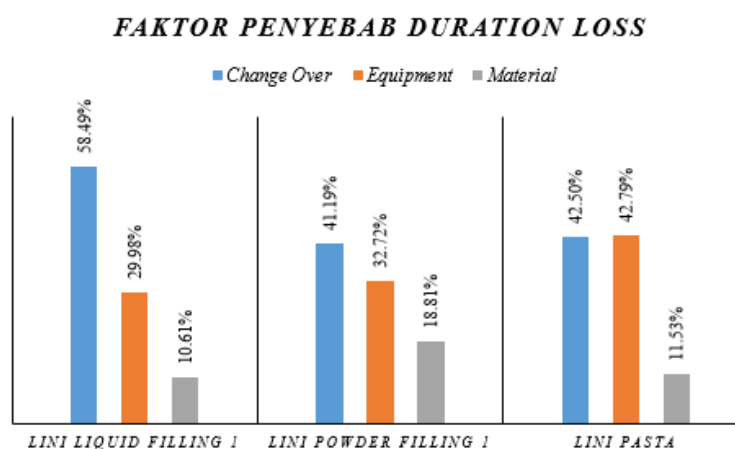
Tahun	Bulan					
	<i>Jan</i>	<i>Feb</i>	<i>Mar</i>	<i>Apr</i>	<i>May</i>	<i>Jun</i>
2015	60.80%	60.70%	58.71%	67.11%	62.00%	58.00%
2016	62.25%	66.61%	62.66%	66.40%	62.23%	62.36%
2017	59.92%	63.26%	68.14%	56.81%	63.96%	62.38%
2018	63.87%	65.35%	61.68%	65.28%	67.68%	-

Tabel 1. 1 Nilai OEE tahun 2015 – 2018 (lanjutan)

Tahun	Bulan					
	Jul	Aug	Sep	Oct	Nov	Dec
2015	63.48%	66.55%	57.05%	60.98%	62.92%	54.18%
2016	53.60%	54.37%	54.77%	57.09%	59.62%	58.86%
2017	61.24%	55.16%	53.09%	58.97%	62.58%	62.37%
2018	-	-	-	-	-	-

Berdasarkan Tabel 1.1 mengenai hasil OEE dari tahun 2015 hingga 2018 dapat dilihat bahwa nilai OEE tertinggi berada pada nilai 68%. Hal ini berarti nilai OEE PT Bayer Indonesia Divisi *Cropscience* masih berada jauh di bawah standar *world class OEE*. Nilai OEE yang masih rendah disebabkan karena nilai *availabilitas* yang masih rendah sedangkan nilai *quality rate* di atas 95% dan nilai *performance efficiency* di atas 80%. Hingga saat ini, nilai *availabilitas* paling tinggi berada pada angka 70%.

Availabilitas merupakan hasil pembagian waktu aktual dengan waktu teoritis. Waktu aktual merupakan selisih antara waktu teoritis dengan *duration loss* (waktu yang hilang dan tidak dapat melakukan proses produksi). Berdasarkan database internal PT Bayer Indonesia Divisi *Cropscience* pada bulan Januari hingga Mei 2017, *total duration loss* salah satu lini produksi mencapai 81 jam yang berarti PT Bayer Indonesia Divisi *Cropscience* kehilangan hampir 11 *shift* untuk melakukan proses produksi. Tentunya *total duration loss* untuk keseluruhan lini akan menjadi sangat besar dalam satu tahun. Faktor utama penyebab *duration loss* ditunjukkan pada Gambar 1.2 berikut:



Gambar 1. 2 Faktor Penyebab *Duration Loss* pada Lini *Liquid, Powder, Pasta*

Berdasarkan Gambar 1.2 dapat dilihat bahwa penyebab terbesar *duration loss* adalah *change over*. *Change over* terjadi ketika terdapat proses *setup* (pergantian produk dan pergantian ukuran produk yang hendak diproduksi), *minor adjustment* setelah proses *setup* atau ketika proses produksi tengah berjalan, dan lain-lain. Sedangkan *duration loss* terkait *equipment* berhubungan dengan kerusakan mesin. *Duration loss* kategori *material* terjadi ketika proses produksi terhambat karena *density material* yang digunakan berbeda, kualitas material yang berbeda-beda dan sebagainya. *Change over* sebagai faktor utama perlu diperhatikan sehingga dapat mengurangi total dari *duration loss*. Salah satu hal yang dapat dilakukan adalah mengurangi proses *setup*. Hal ini dapat dilakukan dengan cara menentukan penjadwalan produksi yang tepat sedemikian hingga proses *setup* tidak sering dilakukan. *Setup time* yang ada termasuk dalam *sequence dependent setup* yang berarti urutan produk yang dijadwalkan untuk diproduksi mempengaruhi lamanya proses *setup*.

Selama ini proses penjadwalan produksi dilakukan secara manual yaitu dari MPS kemudian melakukan proses *plotting* jenis produk yang hendak diproduksi ke dalam lini yang tersedia. Setelah selesai, maka dilakukan rapat mingguan untuk mendiskusikan urutan penjadwalan operasi produksi. Dalam melakukan proses penjadwalan produksi, pihak departemen produksi tidak mempertimbangkan waktu *setup* sesungguhnya akibat pergantian produk yang terjadi. Padahal waktu *setup* mempengaruhi *total duration loss* kemudian *total duration loss* akan mempengaruhi availabilitas yang mana merupakan elemen dari OEE. Tentunya penjadwalan dengan pertimbangan *sequence dependent setup time* menjadi sulit dilakukan secara manual karena variasi produk yang tinggi yaitu lebih dari 80 variasi produk.

Penjadwalan dilakukan untuk 8 lini yaitu 4 lini paralel yang bersifat *identic* untuk produk *powder* dan 4 lini yang bersifat *uniform* untuk produk *liquid*. Dikatakan *identic parallel* ketika kecepatan semua lini dalam memproduksi suatu produk sama sedangkan *uniform parallel* ketika kecepatan antar lini dalam melakukan proses produksi tidak sama. Ketika lini bersifat *uniform parallel* maka perlu mempertimbangkan untuk mengalokasikan produk dengan permintaan produksi tinggi ke lini yang memiliki kecepatan tinggi. Hal ini bertujuan untuk

mencegah total *processing time* yang panjang yang dapat berakibat pada keterlambatan produksi dari *deadline*. Kedua sifat lini paralel ini menjadi pertimbangan selain *sequence dependent setup time*. Selain itu kedua jenis lini ini juga mempertimbangkan *machine eligibility restriction*. Hal ini berarti lini tidak dapat memproduksi semua variasi produk melainkan setiap lini memiliki karakteristik terhadap produk apa yang dapat diproduksi. Dalam melakukan penjadwalan perlu memperhatikan *machine eligibility restriction* yang ada sehingga proses produksi dapat berjalan sesuai tujuan yang diinginkan.

Penjadwalan produk yang dapat dilakukan pada awal bulan hanya produk yang bersifat *dependent* atau MTS sedangkan produk yang bersifat *independent* atau MTO baru dapat dijadwalkan setelah produk tersebut datang. Produk MTO mendapatkan prioritas untuk harus segera diproduksi setelah produk datang. Kedatangan produk di tengah-tengah waktu berjalan ini mengharuskan penjadwalan di awal bulan dilakukan dengan baik sehingga masih ada sisa kapasitas produksi untuk memproduksi produk MTO. Hal ini berarti PT Bayer Indonesia Divisi *Cropscience* berusaha untuk meminimasi *makespan* sehingga pada saat bulan berjalan produk MTO yang datang dapat segera diproduksi karena kapasitas yang masih tersedia. *Makespan* merupakan *completion time* yang paling maksimal dari lini yang tersusun secara paralel. Selain itu, minimasi *makespan* juga bertujuan untuk meminimasi terjadinya proses produksi melebihi waktu produksi yang seharusnya yang menyebabkan perusahaan mengeluarkan biaya untuk pekerja *overtime* agar dapat memenuhi permintaan produksi. Tujuan minimasi *makespan* pada lini *parallel* ini juga dapat memberikan efek berupa seimbangnnya beban produksi antara lini yang satu dengan lini yang lainnya sehingga proses produksi dapat segera diselesaikan.

Penelitian terkait mesin paralel dengan tujuan minimasi *makespan* telah banyak dilakukan. Chuang, et al. (2010) melakukan penelitian pada mesin paralel tidak identik dengan tujuan minimasi *total completion time* dengan pertimbangan *sequence dependent setup* dan *machine preferency*. Huang dan Yu (2010) melakukan penelitian untuk meminimasi *makespan* pada mesin paralel identik dengan pertimbangan *machine eligibility restriction*. Centeno & Armacost (2013) melakukan penelitian dengan tujuan minimasi *makespan* pada mesin paralel identik

dengan mempertimbangkan *release time* dan *machine eligibility restriction*. Kuo, et al. (2017) melakukan penelitian untuk meminimasi *makespan* dengan pertimbangan *sequence dependent setup time*.

Permasalahan pada PT Bayer Indonesia Divisi *Cropscience* yaitu penjadwalan produksi pada lini paralel identik dan *uniform* dengan pertimbangan *sequence dependent setup time*, *machine eligibility restriction* dan *release time* dengan tujuan meminimasi *total setup time* dan *makespan*. Berdasarkan kondisi objek amatan dan penelitian terdahulu, pendekatan heuristik mampu mempertimbangkan seluruh pertimbangan yang telah disebutkan dengan baik. Oleh karena itu, penelitian ini akan mengembangkan pendekatan heuristik yang akan terbagi menjadi dua fase yaitu *constructive* dan *improvement*.

1.2 Rumusan Masalah

Berdasarkan latar belakang di atas, maka dapat ditarik rumusan permasalahan yaitu: bagaimana mengembangkan algoritma penjadwalan pada lini paralel identik dan *uniform* dengan pertimbangan *sequence dependent setup time*, *machine eligibility restriction* dan *release time* dengan tujuan meminimasi *makespan* dan *total duration loss* studi kasus pada PT Bayer Indonesia Divisi *Cropscience*.

1.3 Tujuan

Pada subbab ini akan dijelaskan mengenai tujuan dilakukannya penelitian ini sebagai berikut:

1. Mengembangkan algoritma yang mempertimbangan *sequence dependent setup time*, *release time* dan *machine eligibility restriction* sehingga dapat menyelesaikan permasalahan di PT Bayer Indonesia Divisi *Cropscience*.
2. Meminimasi *total duration loss* dan *makespan* guna memperbaiki nilai availabilitas.
3. Melakukan perbandingan antara penjadwalan sistem sesungguhnya dengan penjadwalan menggunakan algoritma yang telah dikembangkan.
4. Mampu memberikan rekomendasi penjadwalan dan melakukan perbandingan pada operasi produksi *peak season* dan *normal season*.

1.4 Manfaat

Pada subbab ini akan dijelaskan mengenai manfaat dilakukannya penelitian ini sebagai berikut:

1. Menjadi rekomendasi bagi *production planner* dalam melakukan proses penjadwalan operasi produksi sehingga dapat meningkatkan nilai OEE.
2. Menjadi rujukan dalam penelitian yang berhubungan dengan penjadwalan operasi produksi pada mesin paralel dengan pertimbangan *sequence dependent setup*, *machine eligibility restriction* dan *release date*.

1.5 Batasan Penelitian

Pada subbab ini akan dijelaskan mengenai batasan yang akan digunakan dalam penelitian ini yaitu sebagai berikut:

1. Data yang digunakan adalah data dari tahun 2014 hingga 2018.
2. Penjadwalan hanya dilakukan untuk produk *liquid* dan *powder*.
3. Penjadwalan operasi produksi dilakukan pada *peak season* yaitu pada bulan Maret-April serta *normal season* yaitu pada bulan September-Oktober.

1.6 Asumsi Penelitian

Pada subbab ini akan dijelaskan mengenai asumsi yang digunakan dimana dalam penelitian yaitu sebagai berikut:

1. Bahan baku pembuat pestisida selalu tersedia.
2. Mesin tidak mengalami *breakdown*.
3. *Days available* setiap bulan adalah 20 hari.
4. Tidak diperbolehkan adanya interupsi pada proses produksi yang sedang berjalan.

1.7 Sistematika Penulisan

Pada subbab ini akan dijelaskan mengenai sistematika penulisan dalam penelitian mulai dari pendahuluan hingga penarikan kesimpulan dan saran.

BAB 1 PENDAHULUAN

Dalam bab ini akan dijelaskan mengenai latar belakang dilakukannya penelitian ini yang kemudian akan dilanjutkan dengan rumusan masalah. Selain itu juga terdapat tujuan dan manfaat dilakukannya penelitian ini. Dicantumkan pula batasan dan asumsi yang digunakan dalam penelitian.

BAB 2 TINJAUAN PUSTAKA

Dalam bab ini akan dijelaskan mengenai dasar-dasar teori yang digunakan penulis dalam mengerjakan penelitian. Dasar teori ini membantu penulis untuk memahami lebih jauh mengenai topik atau permasalahan yang dibahas dalam penelitian ini. Dasar teori yang digunakan dapat bersumber dari jurnal, tugas akhir ataupun penelitian terdahulu lainnya yang memiliki topik sejenis.

BAB 3 METODOLOGI PENELITIAN

Dalam bab ini akan dijelaskan mengenai alur atau langkah-langkah yang dilakukan penulis dalam menyelesaikan penelitian ini. Langkah-langkah ini ditampilkan dalam *flowchart* sehingga dapat dengan mudah dipahami.

BAB 4 PENGUMPULAN DATA DAN PERANCANGAN ALGORITMA

Pada bab ini akan dijelaskan mengenai data primer dan sekunder yang diperlukan penulis dalam menyelesaikan penelitian ini. Data ini akan menjadi input untuk proses pengolahan data. Data dapat diambil langsung dari PT Bayer Indonesia Divisi *Cropscience* ataupun dari literatur-literatur penunjang. Selanjutnya akan dijelaskan mengenai perancangan algoritma yang digunakan untuk menyelesaikan permasalahan penjadwalan. Kemudian algoritma akan melewati tahap verifikasi dan validasi sebelum akhirnya dimasukan data uji pada bab 5.

BAB 5 UJI COBA DAN ANALISIS

Pada bab ini akan dijelaskan mengenai penerapan model yang telah dikembangkan pada bab sebelumnya dengan menggunakan data yang telah diambil dari PT Bayer Indonesia Divisi *Cropscience*. Selain itu juga akan dilakukan analisis sensitivitas terhadap model yang telah dikembangkan pada bab 5.

BAB 6 KESIMPULAN DAN SARAN

Pada bab ini akan dijelaskan mengenai kesimpulan dari penelitian ini dan juga saran yang dapat diberikan untuk PT Bayer Indonesia Divisi *Cropscience* ataupun untuk penelitian berikutnya.

(Halaman ini sengaja dikosongkan)

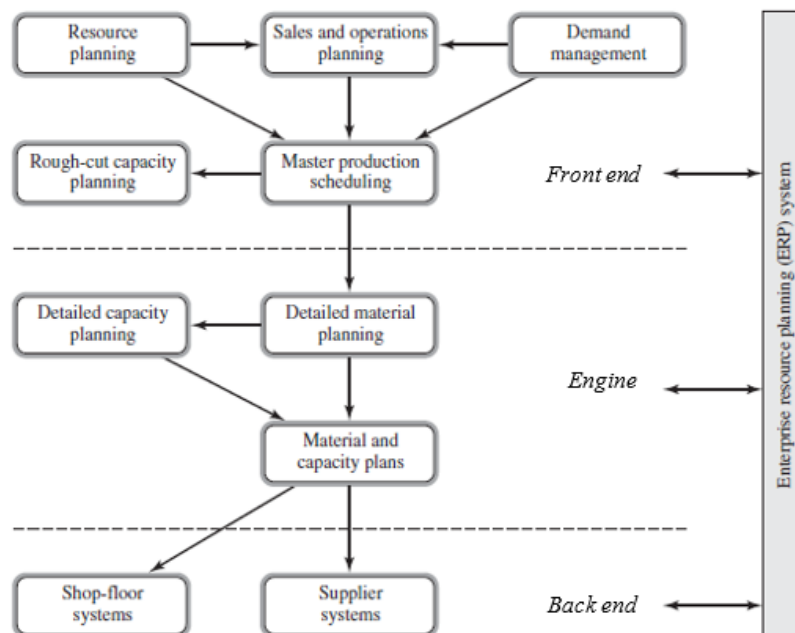
BAB 2

TINJAUAN PUSTAKA

Pada bab ini dijelaskan tentang teori-teori yang berkaitan dengan penelitian tugas akhir. Selain itu juga akan dijelaskan mengenai penelitian terdahulu.

2.1 *Manufacturing Planning and Control*

Penjadwalan merupakan suatu hal yang tidak dapat berdiri sendiri. Perlu adanya integrasi dari fungsi-fungsi lainnya dimana integrasi ini memberikan input data sebelum penjadwalan dilakukan. Penjadwalan operasi sebelum dilakukan perlu menunggu *Master Production Scheduling* (MPS) selesai dibuat kemudian proses *Rough Cut Capacity Planning* (RCCP). Keseluruhan mengenai hal ini berada dalam *manufacturing planning and control* (MPC). MPC berfokus pada perencanaan dan pengendalian segala aspek dalam manufaktur termasuk mengelola material, penjadwalan mesin dan tenaga kerja serta melakukan koordinasi suppliers dengan *key customers*. Terdapat tiga tahapan dalam MPC ini yaitu *planning*, *operations* dan *control*. Berikut adalah skema dari MPC:



Gambar 2. 1 Skema *Manufacturing Planning Control*

Sumber: Fogarty, et al. (1991)

Berdasarkan Gambar 2.1 dapat dilihat bahwa MPC diawali dengan *Sales and Operation Planning* (SOP) dimana SOP berkaitan dengan proses perencanaan yang masih bersifat agregat dengan tujuan mampu menyeimbangkan antara permintaan dan penyediaan. SOP mendapatkan input pula dari *resource planning* untuk mengetahui apakah perusahaan memiliki *resource* yang cukup untuk memproduksi kebutuhan ke depan. Selain itu juga mendapatkan input dari *demand management* yang memberikan informasi mengenai permintaan.

Master Production Scheduling (MPS) merupakan versi *disaggregate* dari SOP dimana dalam MPS terjawab mengenai berapa dan kapan sebuah produk harus diproduksi. Sebelum MPS disetujui dan berlanjut ke tahap berikutnya maka perlu dilakukan pengecekan apakah perusahaan mampu memproduksi sesuai dengan yang dijadwalkan oleh MPS atau tidak. Proses ini merupakan *Rough Cut Capacity Planning* (RCCP). Apabila proses RCCP tidak dilakukan dengan baik maka target produksi yang telah direncanakan dalam MPS bisa tidak terpenuhi. Hal ini karena kapasitas perusahaan ternyata tidak mampu memproduksi sesuai dengan perencanaan yang tercantum dalam MPS. Maka dari itu proses RCCP dilakukan untuk memperbaiki MPS apabila ditemukan kapasitas perusahaan tidak mencukupi.

Kemudian MPS yang telah dilakukan proses RCCP menjadi input untuk *detailed material planning* dan *Master Requirement Planning* (MRP). Dua hal ini berhubungan dengan berapa jumlah material atau bahan baku yang diperlukan untuk memproduksi produk dan kapan material atau bahan baku tersebut harus dipesan atau dibuat maka dari itu dalam membuat MRP perlu untuk mengetahui *Bill of Material* (BOM) dari suatu produk. Terdapat pula *Detailed Capacity Plan* atau biasa dikenal dengan *Capacity Requirement Planning* (CRP) yang mana berfungsi untuk menentukan berapa kapasitas yang diperlukan untuk memenuhi kebutuhan sesuai dengan yang telah ditetapkan dalam MRP. Kapasitas yang diperlukan ini kemudian dibandingkan dengan kapasitas yang tersedia. *Corrective action* dilakukan apabila terdapat perbedaan seperti misalnya melakukan kerja lembur, bekerja sama dengan subkontrak dan sebagainya. Setelah itu maka turun menuju *shop floor system* yang berhubungan dengan penjadwalan operasi produksi produk yang bervariasi pada mesin atau stasiun kerja yang tepat.

2.2 Pengertian Penjadwalan Operasi Produksi

Penjadwalan merupakan permasalahan umum yang kerap terjadi dalam kegiatan produksi dan memberikan dampak terhadap performansi dari penjadwalan itu sendiri seperti misalnya ketidaktercapaian target produksi, tingginya waktu yang dilakukan untuk proses *setup* dan sebagainya. Penjadwalan penting untuk dilakukan dalam upaya mengatur segala sesuatu agar dapat berjalan sesuai dengan target yang diinginkan. Salah satu contoh dari penjadwalan adalah penjadwalan operasi pada suatu lini produksi dengan bentuk penjadwalan berupa susunan urutan produk yang harus diproduksi pada *resource* tertentu. Menurut Baker (1974), penjadwalan merupakan kegiatan mengalokasikan *resource* yang dimiliki untuk menyelesaikan pekerjaan yang ada dimana proses pengurutan pekerjaan dalam 1 jenis *resource* atau yang biasa disebut dengan *sequencing* merupakan salah satu bentuk dari permasalahan penjadwalan yang paling mudah. Kemudian selanjutnya Baker & Trietsch (2009) mengkategorikan model penjadwalan sesuai dengan konfigurasi *resource* serta jenis pekerjaannya. Jika jumlah *resource* adalah satu maka pekerjaan dapat dikategorikan sebagai *single stage* sedangkan jika lebih dari satu maka dikategorikan sebagai *multiple stage*. Jenis pekerjaan dibagi menjadi dua yaitu statis dan dinamis. Dikatakan statis apabila pekerjaan tersedia untuk penjadwalan dan tidak berubah seiring dengan berjalannya waktu. Sedangkan dikatakan dinamis apabila terdapat pekerjaan baru yang muncul seiring dengan berjalannya waktu.

Selain itu menurut Pinedo (2008), penjadwalan merupakan proses pengalokasian *resource* untuk mengerjakan sebuah pekerjaan dalam periode waktu tertentu dan berusaha untuk mencapai satu tujuan ataupun lebih. Penjadwalan juga merupakan proses pengambilan keputusan dasar yang ada di perusahaan manufaktur ataupun industri.

2.3 Klasifikasi Penjadwalan Operasi Produksi

Pinedo (2008) juga menjelaskan bahwa permasalahan penjadwalan dideskripsikan dalam *triplet* $\alpha \mid \beta \mid \gamma$. Dimana α menjelaskan mengenai *machine environment* dengan penjelasan di bawah ini:

1. *Single machine*

Terdiri atas mesin tunggal dengan sejumlah j pekerjaan yang harus diselesaikan. Dimana mesin hanya bisa memproses satu pekerjaan dalam satu waktu dan ketika satu pekerjaan telah dimulai maka tidak ada interupsi dalam proses penyelesaiannya.

2. *Parallel machine*

a. *Identical machines in parallel*

Terdapat sejumlah m mesin paralel identik dimana pekerjaan j diselesaikan oleh salah satu dari m mesin yang ada.

b. *Machines in parallel with different speeds(Uniform)*

Terdapat sejumlah m mesin paralel dengan kecepatan yang berbeda-beda dimana waktu proses untuk menyelesaikan pekerjaan j di mesin m merupakan hasil bagi lamanya proses penyelesaian dengan kecepatan mesin m .

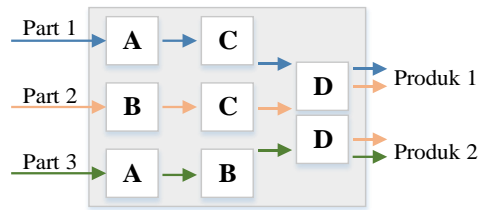
c. *Unrelated machines in parallel*

Dikatakan *unrelated* karena diasumsikan tidak ada relasi antara waktu proses penyelesaian pekerjaan j dengan mesin manapun. Apabila waktu proses penyelesaian pekerjaan di semua mesin sama maka dapat dikatakan bahwa mesinnya identik (Vazirani, 2003).

Kemudian β berisi penjelasan tentang karakteristik proses dan konstrain yang mungkin ada dalam permasalahan penjadwalan. Konstrain yang memungkinkan seperti misalnya *release date*, *preemption*, *job families*, *sequence dependent setup times*, *sequence independent setup times*, *batch processing* dan sebagainya. Berikut adalah karakteristik proses yang dijelaskan dalam β :

1. *Flowshop*

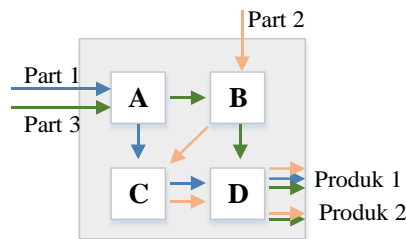
Terdiri dari m mesin dimana setiap *job* harus diproses pada tiap salah satu m mesin tersebut. Setiap pekerjaan mengikuti pola proses/rute yang sama seperti misalnya jika pekerjaan 1 melalui mesin 1 kemudian 2 dan 3 maka pekerjaan yang lain juga mengikuti pola tersebut. *Resource* atau mesin disusun berdasarkan urutan dari proses operasi. Berikut adalah ilustrasi dari *flow shop*:



Gambar 2. 2 *Flow shop Process*

2. *Job shop*

Berbeda dengan *flowshop*, pada *job shop* pola/rute aliran produksi berbeda untuk setiap pekerjaan. Masing-masing pekerjaan memiliki urutan yang beda sehingga tidak selalu menuju ke satu arah. Berikut adalah ilustrasi dari *job shop*:



Gambar 2. 3 *Job Shop Process*

3. *Open shop*

Terdapat sejumlah m mesin dimana setiap pekerjaan harus diproses pada suatu set stasiun kerja pada waktu yang telah diberikan. Pembuat jadwal diperbolehkan untuk menentukan rute proses yang berbeda untuk setiap pekerjaan.

Sedangkan γ menjelaskan tentang *objective* atau tujuan yang hendak dicapai dalam proses penjadwalan seperti misalnya meminimasi waktu penyelesaian pekerjaan, *tardiness*, *makespan* ataupun *objective* lainnya yang dapat disesuaikan dengan jenis penjadwalan yang sedang dilakukan.

2.4 *Setup Time*

Waktu *setup* adalah waktu yang digunakan untuk menyiapkan lini produksi ketika hendak melakukan proses produksi. Menurut Abdallah & Jang (2017), jenis waktu *setup* dibagi menjadi dua yaitu *sequence-independent setup* dan *sequence-dependent setup*. *Sequence-independent setup* merupakan jenis *setup* yang

tergantung pada pekerjaan yang akan dilakukan setelahnya. Sedangkan untuk *sequence-independent setup* artinya waktu *setup* bergantung pada pekerjaan yang disedang dilakukan sekarang dan pekerjaan yang sebelumnya. Pinedo (1995) menyebutkan bahwa proses *setup* pada mesin dapat memakan availabilitas mesin lebih dari 20% apabila tidak diatur secara baik.

2.5 Literature Review

Dalam subbab ini akan dijelaskan mengenai penelitian yang pernah dilakukan terkait penjadwalan operasi produksi dengan mempertimbangkan *setup time*, *machine eligibility restriction*, *uniform* dan *identic parallel machine*.

2.5.1 Single Machine Group Scheduling with Family Setups to Minimize Total Tardiness

Pada penelitian yang dilakukan oleh Gupta & Chantaravarapan (2007) berfokus pada penjadwalan *job* dengan *family* tertentu dengan mempertimbangkan *independent setup* yang muncul ketika terjadi pergantian *family*. Dalam penelitian ini dilakukan hanya untuk satu mesin dengan tujuan meminimasi *job* yang terlambat dari *due date* karena akan ada *penalty* jika terjadi keterlambatan. Terdapat dua fase yang digunakan yaitu *constructive* (yang terdiri dari dua tahap) dan fase *improvement* dengan cara menerapkan *Simulated Annealing*. Pada jurnal ini juga dikembangkan model *mixed-integer programming model* yang dapat menyelesaikan permasalahan penjadwalan *single machine* dengan *family setups*. Berikut adalah notasi yang digunakan:

f	: jumlah <i>family</i> i hingga f
n	: jumlah <i>job</i> yang ada dalam <i>family</i> i yang <i>available</i> saat <i>time zero</i>
P_{ij}	: <i>processing time</i> <i>job</i> ke- j dari <i>family</i> i
D_{ij}	: <i>due date</i> <i>job</i> ke- j dari <i>family</i> i
S_i	: <i>setup time</i> untuk <i>family</i> i selanjutnya
T_{ij}	: <i>tardiness</i> <i>job</i> ke- j dari <i>family</i> I
$T_{ij} = \max\{0, C_{ij} - D_{ij}\}$	
C_k	: <i>completion time</i> untuk <i>job</i> urutan ke- k

- T_k : *total tardiness* untuk *job* urutan ke- k
 S : *scheduled family*
 U : *unscheduled family*

Pendekatan algoritma *heuristic* diharapkan mampu mencari nilai optimal atau *near-optimal* dengan CPU *time* lebih cepat. Pendekatan yang ditawarkan terdiri atas 2 fase dimana fase pertama adalah untuk men-*generate initial schedule* sedangkan fase kedua dilakukan untuk meng-*improve initial solution* ini. Fase pertama dilakukan untuk menentukan urutan *job* dalam tiap *family* dan urutan *family groups* nya dimana hal ini dilakukan dengan menggunakan *greedy algorithm*.

2.5.1.1 Fase Penentuan Urutan Job dalam Family

Fase ini dikembangkan dengan menggunakan *Net Benefit of Ratio* (NBR) heuristik. NBR dikenal sangat *useful* dalam mengatasi penjadwalan dengan tujuan minimasi total *tardiness*. Dimana konsep dari NBR adalah melakukan relokasi *job* urutan paling akhir sehingga dapat mengurangi *total tardiness* Berikut adalah langkah-langkah dalam NBR:

1. Hitung *slack* dan *tardiness* untuk setiap *job*.

$$S = \text{Slack of Job} = \max(D_j - C_j, 0)$$

2. Identifikasi setiap *job* yang memiliki *processing time* dan *due date* kurang atau sama dengan *job* lainnya. Apabila terdapat *job* demikian maka *job* yang *processing time* dan *due date* nya lebih rendah akan menjadi *preceding job* tersebut.
3. Untuk *job* urutan paling akhir beri Tag 1 dimana *job* tersebut adalah *job* yang dapat dikurangi nilai *tardiness*-nya.
4. Pilih *job* yang memungkinkan untuk mengganti posisi *job* yang paling akhir. Untuk setiap kemungkinan *job*, hitung:

$$CR_j = \max\{0, \sum_{i=j+1}^k P_i - S_i\}$$

5. Lakukan perhitungan terhadap *benefit gain ratio*:

$$BR_j = \sum_{i=j+1}^k \min\{P_j, T_i\}$$

6. Kemudian hitung NBR dengan rumus:

$$NBR = BR_j - C_j$$

7. *Job* dengan nilai NBR positif terbesar akan terpilih, apabila nilai positif terbesar lebih dari satu maka pilih dengan *processing time* yang lebih tinggi.

2.5.1.2 Fase Penentuan Urutan antar Family Groups

Pada fase ini dilakukan dengan menggunakan *greedy heuristic*. Dalam tahapan ini akan dihitung waktu *setup* yang dapat menyebabkan sebuah *job* mengalami keterlambatan. Berikut ini adalah tahapan algoritmanya:

1. Set $U = \{1, 2, \dots, f\}$ dimana U adalah *unscheduled family* dan $S = \emptyset$ yang berarti masih berupa himpunan kosong (belum ada *family* yang dijadwalkan). Hitung $P = \sum_{k=1}^f [s_k - \sum_{j=1}^{n_k} p_{kj}]$.
2. Pertimbangkan semua *family* yang ada dalam set U dengan menghitung *starting time* nya, $C = P - \sum_{j=1}^{n_k} p_{kj}$ untuk menentukan urutan semua *job* yang ada dalam posisi *family* ke- i . Hitung T_k pada *family* di posisi ke- i .
3. Pilih *family* q dengan total *tardiness* untuk *family* posisi ke- i dan kemudian hapuskan *family* q dari U dan berpindah ke depan urutan *family* yang sudah ada dalam S . Kemudian set $i=i-1$. Jika $i=1$ maka masuk ke tahapan ke-4. Jika tidak maka: $P = P - S_q - \sum_{j=1}^{n_q} p_{kj}$ dan kembali ke tahapan 2.
4. Tentukan urutan dari *job* dari *family* terakhir yang ada dalam U dengan C sebagai proses *setup family* yang terakhir kemudian letakkan *family* tersebut di depan *family* yang sudah ada dalam S .
5. Hitung total *tardiness* dari *final sequence* di S .

2.5.1.3 Fase Simulated Annealing for Family Groups

Fase ini diperlukan untuk meningkatkan kualitas hasil dari fase *constructive* 1 dan 2 yang menggunakan *heuristic*. Hal ini disebabkan karena hasil dari *heuristic* masih terjebak dalam *local optimum*. Maka dari itu pendekatan *metaheuristic* diharapkan mampu memperbaiki *total tardiness* yang telah didapatkan pada fase 3 – *greedy heuristic*. Berikut adalah tahapannya:

1. Tentukan t_{max} , $iter$, t_o , r dan n sebagai *control parameter*. Gunakan S dari hasil *greedy heuristic* sebagai initial schedule.
2. Set *temperature* sekarang $t_{cur} = t_{max}$ dan $q = 1$ dan Z adalah total *tardiness* dari hasil S .
3. J adalah *random number* antara 1 dan f . Masukkan *family j* ke dalam *family position* dan lakukan *reschedule job* yang ada dalam *family* yang terpengaruh karena adanya penambahan ini. Dari semua penjawalan yang di-generate i dari proses *insertion* ini maka pilih urutan *family* dengan *lowest total tardiness* (Z'). Dengan S' adalah *schedule* terpilih dan $\Delta Z = Z' - Z$. Jika $\Delta Z \leq 0$ maka set $S = S'$ dan $Z = Z'$ dan menuju ke step 5. Jika tidak maka menuju ke step 4.
4. Hitung peluang diterimanya urutan yang baru dengan Boltzmann *probability mass function*, $p(c)$. Bangkitkan *random number* antara 0 dan 1, misal h . jika $p(c) > h$ maka terima urutan baru tersebut.
5. Jika $q < iter$, maka set $q = q + 1$ dan kembali ke step 2. Jika tidak maka lanjut ke step 6.
6. Set $t_{cur} = r * t_{cur}$. Jika $t_{cur} > t_o$ maka set $q = 1$ dan kembali ke step 2. Jika tidak maka hentikan proses.

2.5.2 Parallel Machines Scheduling with Preference on Machines

Pada penelitian yang dilakukan oleh Chuang, et al. (2010) membahas mengenai penjadwalan pada perusahaan manufaktur yang bergerak dalam bidang *electro-etching aluminium foil*. Dimana dalam perusahaan ini perlu menjadwalkan *equipment* yang tepat sehingga dapat memproduksi *voltage aluminium foil* dengan *processing time* yang lebih singkat. Hal inilah yang kemudian menyebabkan adanya

machine preference dimana perusahaan cenderung memilih mesin dengan kecepatannya yang tepat untuk menyelesaikan suatu permintaan produk.

Penjadwalan dilakukan pada mesin paralel dengan kecepatan yang berbeda sehingga *processing time* tergantung dari mesin apa yang digunakan untuk produksi. Selain itu juga pertimbangan *sequence dependent setup* digunakan apabila terjadi pergantian *aluminium foil* yang diproduksi. Tujuan dari penelitian ini adalah untuk meminimasi total *completion time* yang dilakukan dengan tiga tahap *heuristic*. Tiga tahap itu adalah *Hungarian method*, heuristik untuk menyelesaikan permasalahan *completion time* pada *parallel machine* dengan *setup times* dan yang terakhir adalah *dynamic programming*. Berikut adalah langkah dari *three-steps heuristic* yang diterapkan dalam Chuang, et al. (2010):

Step 0: Asumsikan *setup time* nol

Step 1: Tambahkan $n \times (m-1)$ *dummy jobs* dengan *zero processing time*

Step 2: Buat matriks $nm \times nm$ dengan isi matriks berupa *processing time*. Lakukan *Hungarian method* untuk menyelesaikan problem ini dan hitung total *completion time* nya dengan mempertimbangkan *setup time*.

Step 3: Jika produk belum di-plot sesuai mesin yang tepat maka lanjut ke step 4

Step 4: Set M sebagai jumlah mesin yang dapat memproses level produk yang sama

Step 4.1: *Job* dengan level (tipe) yang sama dikelompokkan menjadi satu

Step 4.2: Urutkan mesin dari total *processing time* yang paling tinggi hingga yang paling rendah

Step 4.3: Set b_m sebagai *batch job* yang dijadwalkan pada urutan ke m dan $b_{m'}$ sebagai *batch job* yang dijadwalkan pada urutan ke m' . Jika b_m dan $b_{m'}$ berada pada level produk yang berbeda dan $S > |C_m - C_{m'}|$ maka akan diurutkan berdasarkan aturan SPT (*Shortest Processing Time*).

Step 4.4: Jika M berjumlah ganjil maka jumlah $(M+2)/1$ di

Step 4.5: Ulangi step 4.2 hingga 4.4

Step 5: Terapkan *Dynamic Programming* di setiap mesin

Step 6: Atur kembali urutan dalam tiap mesin dan hitung total *completion time* nya dengan mempertimbangkan *property*: “*Job* dengan *processing time* yang panjang

harus diletakkan pada mesin yang sesuai sehingga *processing time* tidak bertambah panjang”.

2.5.3 *Effective Heuristic Considering Machine Flexibility for Parallel Machine with Eligibility Problem*

Huang dan Yu (2010) menjelaskan bahwa permasalahan mesin paralel dengan *eligibility scheduling* pada umumnya terjadi pada perusahaan industri sehingga kemudian dikembangkan algoritma heuristik yang dapat meminimasi *makespan*. Algoritma *Dynamic Flexibility Selection (DFS)* yang dikembangkan adalah algoritma yang mempertimbangkan *job flexibility* dan *machine flexibility*. Berikut adalah tahapannya:

Step 1: Hitung MF (*machine flexibility*) dan JF (*job flexibility*) untuk setiap mesin dan *job*

Step 2: Cari *job* dengan JF paling kecil dan memiliki *processing time* yang paling panjang

Step 3: Cari mesin yang memiliki akumulasi *processing time* terpendek dan memiliki *flexibility* terendah

Step 4: Assign *job* yang terpilih pada step 2 ke mesin yang terpilih pada step 3. Update akumulasi *processing time* tiap mesin.

Step 5: Jika semua *job* telah di-assign maka stop. Jika terdapat *job* belum di-assign maka kembali ke step 1.

Selain itu dalam jurnalnya juga mencantumkan model matematika dengan tujuan *minimize makespan*. *Makespan* pada penjadwalan mesin *parallel* berarti nilai maksimum dari total *processing time* setiap lini. Sedangkan *makespan* pada *single machine* memiliki nilai yang sama dengan total *processing time*. Model matematis ini akan dibandingkan dengan algoritma DFS yang dibuat. Akhir dari jurnal ini menunjukkan bahwa hasil dari algoritma DFS memiliki nilai yang tidak jauh berbeda dengan hasil dari model matematis yang digunakan. Berikut adalah model matematisnya:

Objective Function:

$$\text{Min } Z = C_{\max} \quad (2.1)$$

Subject To:

$$\sum_{j=1}^n X_{ij} Z_{ij} p_j \leq C_{\max} \quad \text{for } i= 1 \dots m \quad (2.2)$$

$$\sum_{i=1}^m X_{ij} Z_{ij} = 1 \quad \text{for } j= 1 \dots n \quad (2.3)$$

Persamaan (2.1) menjelaskan bahwa tujuan utamanya adalah meminimasi *makespan*. *Constraint* (2.2) menjelaskan bahwa *completion time* pada tiap mesin tidak boleh melebihi *makespan*. *Constraint* (2.3) menjelaskan bahwa setiap *job* hanya dapat diproses satu kali.

2.5.4 *Minimizing Makespan on Parallel Machines with Release Time and Machine Eligibility Restriction*

Jurnal penelitian Centeno & Armacost (2013) menjelaskan tentang bagaimana penjadwalan pada mesin parallel dengan batasan dimana tidak setiap mesin mampu memproses semua *job*. Penelitian berkaitan tentang hal ini telah dilakukan sebelumnya oleh Pinedo dengan menggunakan aturan LFJ (*Least Flexible Job*) dimana akan mencapai hasil yang optimal apabila keadaan *machine eligibility restriction* bersifat *nested* dan tidak ada *constraint release time*. Maka dari itu Centeno & Armacost (2013) mengembangkan algoritma yang mampu menjawab keadaan apabila mesin tidak sepenuhnya bersifat *nested* dimana jenis *nested* dari mesin dibagi menjadi 3 yaitu *almost nested case* (60% *nested*) dan *poorly nested case* (20% *nested*) dan juga menjawab *constraint release time job* yang telah diketahui. Dalam jurnalnya dilakukan penggabungan antara *rule* LPT (*Longest Processing Time*), LFJ (*Least Flexible Job*) dan LFM (*Least Flexible Machine*) kemudian dilakukan percobaan terhadap urutan *rule* yang harus dipilih dan dilakukan terlebih dahulu. Hasilnya adalah hanya terdapat perbedaan signifikan ketika LPT (*Least Processing Time*) diterapkan bersamaan dengan LFJ (*Least Flexible Job*) dan LFM (*Least Flexible Machine*) pada permasalahan *poorly nested*

case (20% nested). Sedangkan untuk urutan harus menggunakan LFM (*Least Flexible Machine*) atau LFJ (*Least Flexible Job*) terlebih dahulu tidak memberikan dampak secara signifikan terhadap hasil *makespan* kasus *poorly nested* maupun kasus *almost nested*.

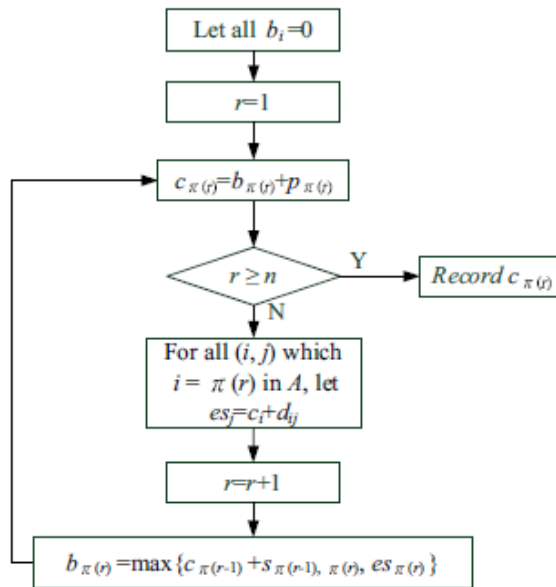
2.5.5 *Single Machine Sceduling with Sequence Dependent Setup times and Delayed Precedence Constraint*

Kuo, et al. (2017) menjelaskan mengenai penjadwalan pada *single machine* dengan mempertimbangkan *sequence dependent setup time* dan *delayed precedence constraint*. *Precedence constraint* berarti terdapat *job* yang harus dikerjakan terlebih dahulu agar *job* selanjutnya dapat dikerjakan sehingga setiap kali penjadwalan dilakukan, urutan dari *job* harus memenuhi aturan ini. Tahap yang dilakukan adalah menghitung *makespan* terlebih dahulu lalu kemudian hasil akan diperbaiki dengan menggunakan *Variable Neighborhood Search (VNS)*. Berikut adalah *makespan calculation model* yang dikembangkan:

b_j	: Starting time dari <i>job j</i>
S_{ij}	: Setup time dari <i>job i</i> ke <i>job j</i>
p_i	: Processing time dari <i>job i</i>
C_i	: Completion time dari <i>job i</i>
d_i	: Delay time dari <i>job i</i>
X_{ij}	: Bernilai 1 jika <i>job j</i> diproses setelah <i>job i</i>
π_r	: Job dengan sequence r
es_j	: Earliest starting time dari <i>job j</i>

Ketika *job i* adalah *precedence* dari *job j* maka *starting time* dari *job j* harus lebih besar dari *completion time job i* ditambah dengan *delay time job i*. Rumus $b_j \geq c_i + d_i$ menjamin bahwa *job j* akan dikerjakan setelah *job i* selesai dilakukan. S_{ij} akan ditambahkan setiap kali *job j* dikerjakan setelah *job i*. Dalam *makespan calculation model* yang dikembangkan diawali dengan memulai nilai $b_i = 0$ dan set $r = 1$. Kemudian hitung C pada urutan ke π dengan menambahkan

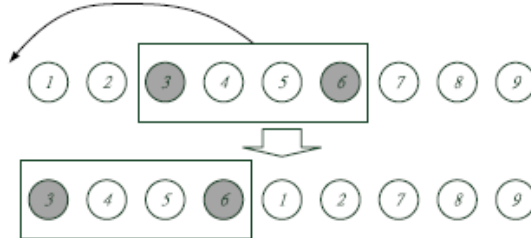
starting time dengan processing time nya. Jika r masih kurang dari n maka akan mencari *job* berikutnya dengan menetapkan bahwa $es_j \geq c_i + d_{ij}$ yang berarti *job* berikutnya paling cepat dapat diproses setelah *job i* selesai dilakukan ditambah dengan *delay* yang terjadi antara *job i* dan *job j*. Kemudian akan dihitung *makespan*. Berikut adalah *flowchart proposed makespan calculation model* yang dibuat oleh Kuo, et al. (2017)



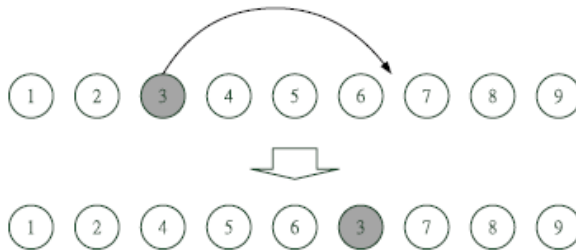
Gambar 2. 4 *Makespan Calculation Model*

Hasil penjadwalan awal ini kemudian akan diperbaiki dengan VNS yang mana merupakan pendekatan *metaheuristic* yang menawarkan perubahan sistematis ke struktur tetangga. VNS akan menggunakan *initial solution* dan akan secara terus menerus mencari solusi dari tetangganya. Namun hal ini memungkinkan *constraint precedence job* dilanggar sehingga perlu dimodifikasi yaitu dengan cara selalu mengecek dalam tiap iterasi apakah *constraint precedence job* terpenuhi atau tidak. Solusi baru dari hasil iterasi berikutnya yang lebih baik (*makespan* yang lebih pendek) akan diterima untuk menggantikan solusi yang sekarang. Namun jika ternyata solusi yang dihasilkan tidak baik maka memungkinkan diterima dengan *acceptance probability* tertentu. Hal ini sama seperti konsep *Simulated Annealing* sehingga perlu menetapkan α sebagai faktor penurunan suhu (T) dengan nilai di antara 0 hingga 1 dengan T adalah *initial*

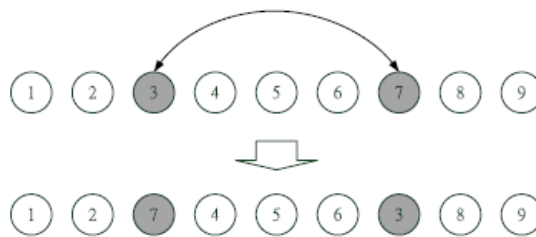
solution dari hasil penjadwalan awal. Dalam VNS ini terdapat 4 operasi yang digunakan untuk mencari ke struktur tetangga yaitu dengan *job exchange*, *job insertion*, *job series move* dan *job series exchange*. Berikut adalah ilustrasi dari masing-masing operasi dalam mencari struktur tetangga:



Gambar 2. 5 *Job Series Move*



Gambar 2. 6 *Job Insertion*



Gambar 2. 7 *Job Exchange*

2.5.6 Penjadwalan Multi Produk pada Mesin Paralel *Filling Lithos*

Pada penelitian yang dilakukan oleh Septiwiarto (2017) membahas mengenai penjadwalan multi produk pada mesin *parallel* dengan jumlah variasi produk mencapai 50. Dalam penelitian ini mempertimbangkan waktu *setup* yang *independent* dalam melakukan penjadwalan. Hal ini disebabkan karena waktu *setup* dapat mengurangi ketersediaan waktu produksi sehingga dapat menyebabkan produksi tidak memenuhi target. Variabel keputusan yang digunakan terkait jumlah

produk yang harus diproduksi dalam satuan waktu yang dinyatakan dalam bilangan *integer*. Objektif yang ingin dicapai adalah memaksimalkan keuntungan yang dapat diterima apabila mampu menjadwalkan proses produksi dengan baik sehingga dapat memenuhi target produksi. Indeks yang digunakan dalam menyelesaikan permasalahan adalah sebagai berikut:

i	: indeks untuk produk
t	: indeks untuk hari pelaksanaan produksi
m	: indeks untuk mesin yang akan digunakan dalam melakukan proses
(i, j)	: indeks untuk perintah pergantian produk i ke produk j
(i, t, m)	: indeks untuk produk i pada hari t dan proses produksi dilakukan pada mesin m
(i, m)	: indeks untuk perintah produk i yang akan diproduksi pada mesin m
(i, j, t, m)	: indeks untuk perintah pergantian produk i ke produk j pada hari t dan proses produksi dilakukan pada mesin m

Berikut adalah notasi yang digunakan untuk menyelesaikan permasalahan yang ada:

WP_{itm}	: Keputusan untuk melaksanakan produksi pada produk i yang diproduksi pada hari t dan diproses dengan menggunakan mesin m
SU_{ij}	: Waktu pergantian <i>setup</i> dari produk i ke produk j
V_{im}	: Kecepatan produksi produk i yang diproses pada mesin m dalam satuan unit per jam
I_{it}	: <i>Inventory</i> produk i pada hari t
S_{it}	: Stok hasil produksi <i>produk</i> i pada hari t
D_i	: Program produksi yang diterima oleh perusahaan untuk setiap produk i
P_i	: Profit yang diterima oleh perusahaan untuk setiap produk i
WS	: Waktu <i>setup</i> awal produksi hari pertama
HPP_{it}	: Beban produksi produk i dan pada hari t yang merupakan data <i>dummy</i>

JSP : Tingkat persentase *demand* produk yang mengalami kondisi *stock out* dan harus diproduksi terlebih dahulu pada hari yang telah ditentukan

WK : Waktu kerja yang tersedia dalam satu hari, ketersediaan waktu produksi yang nantinya akan dikurangi waktu setup

Selanjutnya adalah notasi-notasi yang menunjukkan variabel keputusan dalam penelitian ini:

X_{itm} : Jumlah produk i yang diproduksi pada hari t dan diproses dengan menggunakan mesin m

Y_{itm} : Keputusan untuk melaksanakan produksi pada produk i yang diproduksi pada hari t dan diproses dengan menggunakan mesin m .
 $Y = 1$ jika dilakukan kegiatan produksi
 $Y = 0$ jika tidak dilakukan kegiatan produksi

Z_{ijtm} : Keputusan untuk melakukan pergantian produk atau tetap melakukan produksi pada produk yang sama, saat produk ganti $i \neq i$ pada hari $t+1$ dan diproses dengan menggunakan mesin m .
 $Z = 1$ jika terdapat perpindahan produk saat hari $t+1$.

Model matematis yang digunakan untuk menyelesaikan permasalahan ini adalah sebagai berikut:

Objective Function:

$$Z = \text{Max} \left(\sum_{i=1}^l \sum_{t=1}^T \sum_{m=1}^M X_{itm} \times P_i - \sum_{i=1}^l \sum_{t=1}^T \sum_{m=1}^M X_{itm} \times HPP_{it} \right) \quad (2.4)$$

Subject To:

$$\sum_{i=1}^l \sum_{m=1}^M X_{itm} < D_i \quad \text{for } i=1 \dots i \quad (2.5)$$

$$\sum_{i=1}^l Y_{itm} \leq 1 \quad \text{for } t=1 \dots t, m=1 \dots m \quad (2.6)$$

$$X_{itm} \leq V_{im} * WP_{itm} \quad \text{for } t=1 \dots t, m=1 \dots m, i=1 \dots i \quad (2.7)$$

$$WP_{itm} = WS * Y_{itm} \quad \text{for } t=1, m=1 \dots m, i=1 \dots i \quad (2.8)$$

$$\sum_{i=1}^l Z_{ijtm} * (WK - SU_{ij}) = WP_{itm} \quad \text{for } t=2...t, m=1...m, i=1...i \quad (2.9)$$

$$\sum_{i=1}^l Z_{ijtm} = Y_{itm} \quad \text{for } t=1...t, m=1...m, i=1...i \quad (2.10)$$

$$\sum_{i=1}^l Z_{ijt-1m} = Y_{jtm} \quad \text{for } t=2...t, m=1...m, j=1...j \quad (2.11)$$

$$\sum_{m=1}^M X_{itm} = St_{it} \quad \text{for } m=1...m, i=1...i \quad (2.12)$$

$$I_{it} = St_{it} \quad \text{for } t=1, i=1...i \quad (2.13)$$

$$I_{it} = St_{it} + I_{it-1} \quad \text{for } t=2...t, i=1...i \quad (2.14)$$

$$I_{it} > JSP * D_i \quad \text{for } t=1...t, i=1...i \quad (2.15)$$

$$X_{itm} \in integer \quad \text{for } t=1...t, m=1...m, i=1...i \quad (2.16)$$

$$Y_{itm} \in biner \quad \text{for } t=1...t, m=1...m, i=1...i \quad (2.17)$$

$$Z_{ijtm} \in biner \quad \text{for } t=1...t, m=1...m, i=1...i, j=1...j \quad (2.18)$$

Persamaan (2.4) merupakan fungsi tujuan yang memaksimalkan keuntungan yang akan diterima oleh perusahaan. Dimana merupakan hasil perkalian produk yang berhasil diproduksi dengan besarnya keuntungan kemudian dikurangi dengan total HPP nya. Persamaan (2.5) merupakan konstrain yang memicu adanya perintah produksi produk i yang akan diproduksi pada hari t dan dengan menggunakan mesin m agar dapat memenuhi program yang telah diterima oleh PUG. Persamaan ini menjamin bahwa keseluruhan produksi tidak akan melebihi program produksi yang telah diterima. Persamaan (2.6) merupakan perintah pelaksanaan produksi dimana hanya boleh menggunakan 1 mesin untuk melakukan proses produk. Persamaan (2.7) menjamin bahwa jumlah produk yang akan diproduksi tidak boleh melebihi kapasitasnya dengan mempertimbangkan kecepatan mesin dan ketersediaan waktu. Persamaan (2.8) untuk menentukan waktu produksi pada mesin produksi saat hari pertama pelaksanaan program produksi. Persamaan (2.9) ditentukan oleh persamaan (2.10) dan (2.11) yang melakukan pengecekan apakah produksi hari ini berbeda dengan hari sebelumnya. persamaan (2.12) memberikan keputusan pergantian produk i ke produk j . Persamaan (2.13) hingga (2.14) berguna untuk melakukan *update* dari *inventory* produk yang telah diproduksi.

2.6 Posisi Penelitian

Pada subbab ini akan dijelaskan mengenai posisi penelitian yang dilakukan penulis dengan cara membandingkan penelitian terdahulu dengan keadaan penelitian terkini. Perbandingan penelitian disajikan dalam tabel di bawah ini:

Tabel 2. 1 Posisi Penelitian

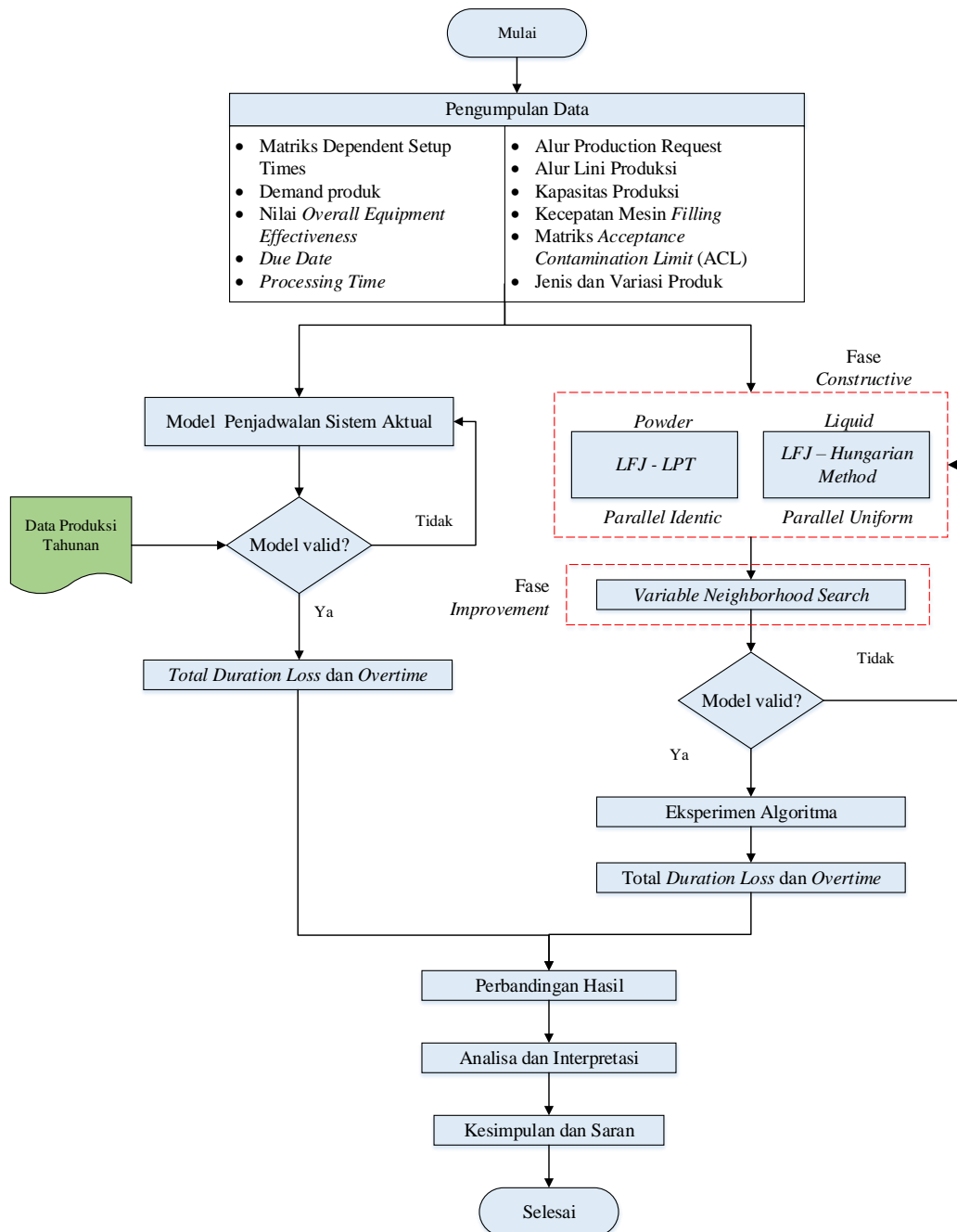
No	Penelitian	Pertimbangan					Metode	Objective
		<i>Uniform Parallel Machine</i>	<i>Identic Parallel Machine</i>	<i>SDST</i>	<i>Release Time</i>	<i>Machine Eligibility Restriction</i>		
1	Gupta & Chantaravarapan (2007)			√			<i>Greedy Algorithm, Net Ratio dan Simulated Annealing</i>	<i>Tardiness</i>
2	Huang dan Yu (2010)		√			√	<i>Dynamic Flexibility Selection (DFS)</i>	<i>Makespan</i>
3	Chuang, et al. (2010)	√		√		√	<i>Hungarian Method, Dynamic Programming and Heuristic Approach</i>	<i>Completion Time</i>
4	Centeno & Armacost (2013)		√		√	√	<i>LFJ, LFM, LPT</i>	<i>Makespan</i>
5	Kuo, et al. (2017)			√			<i>Makespan calculation model dan Variable Neighborhood Search (VNS)</i>	<i>Makespan</i>
6	Septiwiando (2017)		√				<i>ILP</i>	<i>Profit</i>
7	Penelitian ini	√	√	√	√	√	<i>LFJ-LPT, Hungarian Method, Variable Neighborhood Search (VNS), On-line time update</i>	<i>Makespan dan total duration loss</i>

(Halaman ini sengaja dikosongkan)

BAB 3

METODOLOGI PENELITIAN

Pada bab ini akan dijelaskan mengenai tahapan-tahapan dalam melakukan penelitian. Langkah-langkah secara detail dapat dilihat pada *flowchart* di bawah ini:



Gambar 3. 1 Diagram Alir Penelitian

Subbab di bawah ini merupakan penjelasan detail mengenai alur penelitian berdasarkan *flowchart* penelitian yang telah ditentukan.

3.1 Tahap Pengumpulan Data

Dalam tahap yang pertama ini akan ditampilkan data-data yang diperlukan dalam pengerjaan penelitian. Data yang diperlukan antara lain data *demand* atau permintaan produksi, jenis produk yang akan ditampilkan dalam kode komersial produk, variasi ukuran produk, jumlah *days available*, matriks *setup time*, matriks ACL, kapasitas produksi, ilustrasi lini produksi disertai *flow process*-nya. Data-data ini kemudian akan dijadikan input untuk perhitungan dari model yang telah dibuat. Data yang didapat akan ditampilkan pada BAB IV.

3.2 Tahap Pengembangan Model Penjadwalan Sistem Aktual

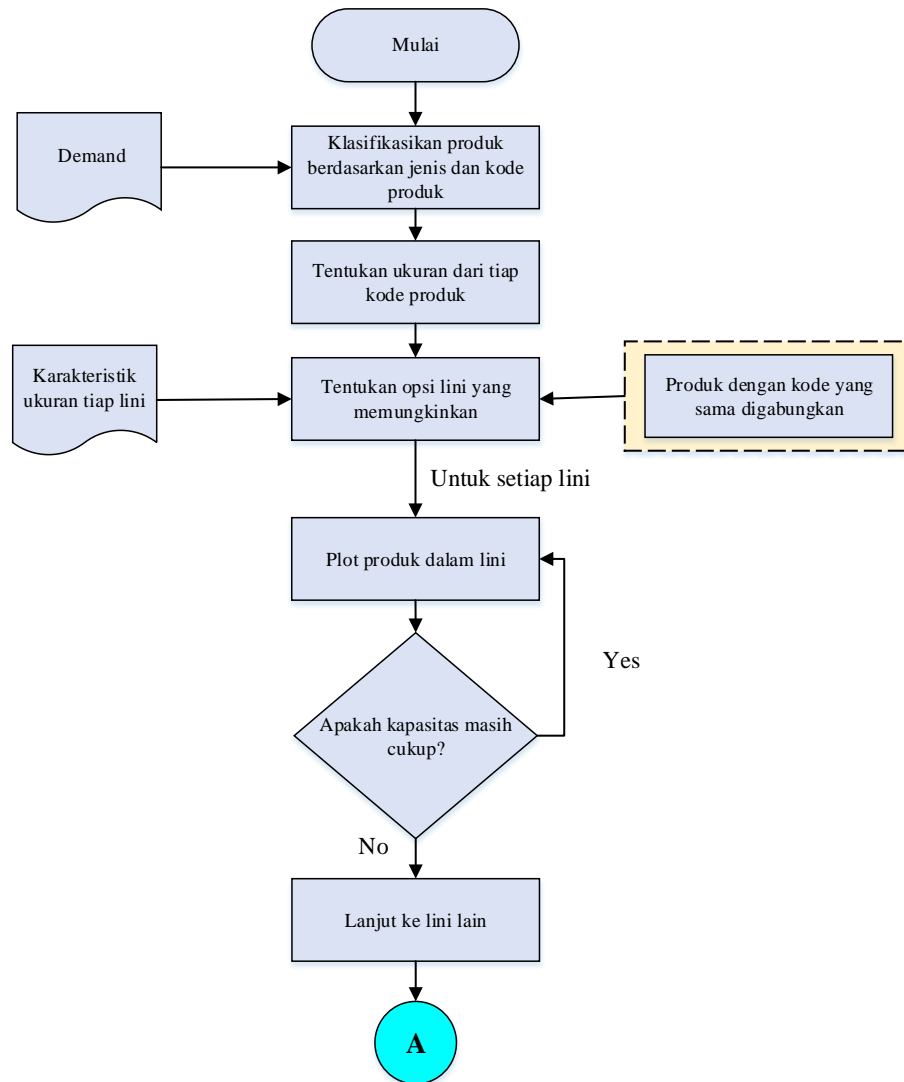
Dalam tahap ini akan dikembangkan model penjadwalan keadaan aktual di PT Bayer Indoensia Divisi *Cropscience*. Proses pengembangan model ini dilakukan dengan cara mengamati proses penjadwalan yang dilakukan langsung oleh *production planner*. Proses penjadwalan eksisting akan terbagi menjadi tiga tahap yaitu penentuan lini, penentuan urutan produk dalam lini dan perhitungan *makespan*

3.2.1 Penentuan Lini

Berdasarkan pengamatan yang dilakukan penulis dan hasil wawancara didapatkan langkah bagaimana melakukan proses penentuan lini apabila terdapat *job* atau produk untuk diproduksi. Langkah yang dilakukan adalah sebagai berikut:

1. Identifikasi ukuran produk yang hendak diproduksi.
2. Tentukan lini mana saja yang dapat memproduksi produk tersebut sesuai dengan ukuran produknya dan karakteristik lininya. Setiap lini memiliki kemampuannya masing-masing untuk memproduksi produk dengan ukuran ukuran tertentu.
3. Produk dengan kode komersial yang sama diprioritaskan untuk dialokasikan pada lini yang sama.
4. Produk yang dialokasikan ke dalam lini harus dicek apakah produk-produk tersebut mampu dipenuhi sesuai dengan *due date* nya atau tidak jika

ditempatkan dalam 1 lini. Jika tidak, maka harus dipindah ke lini lain sehingga tidak melebihi *due date*. Langkah-langkah tersebut ditampilkan *flowchart* model konseptual penentuan lini di bawah ini:

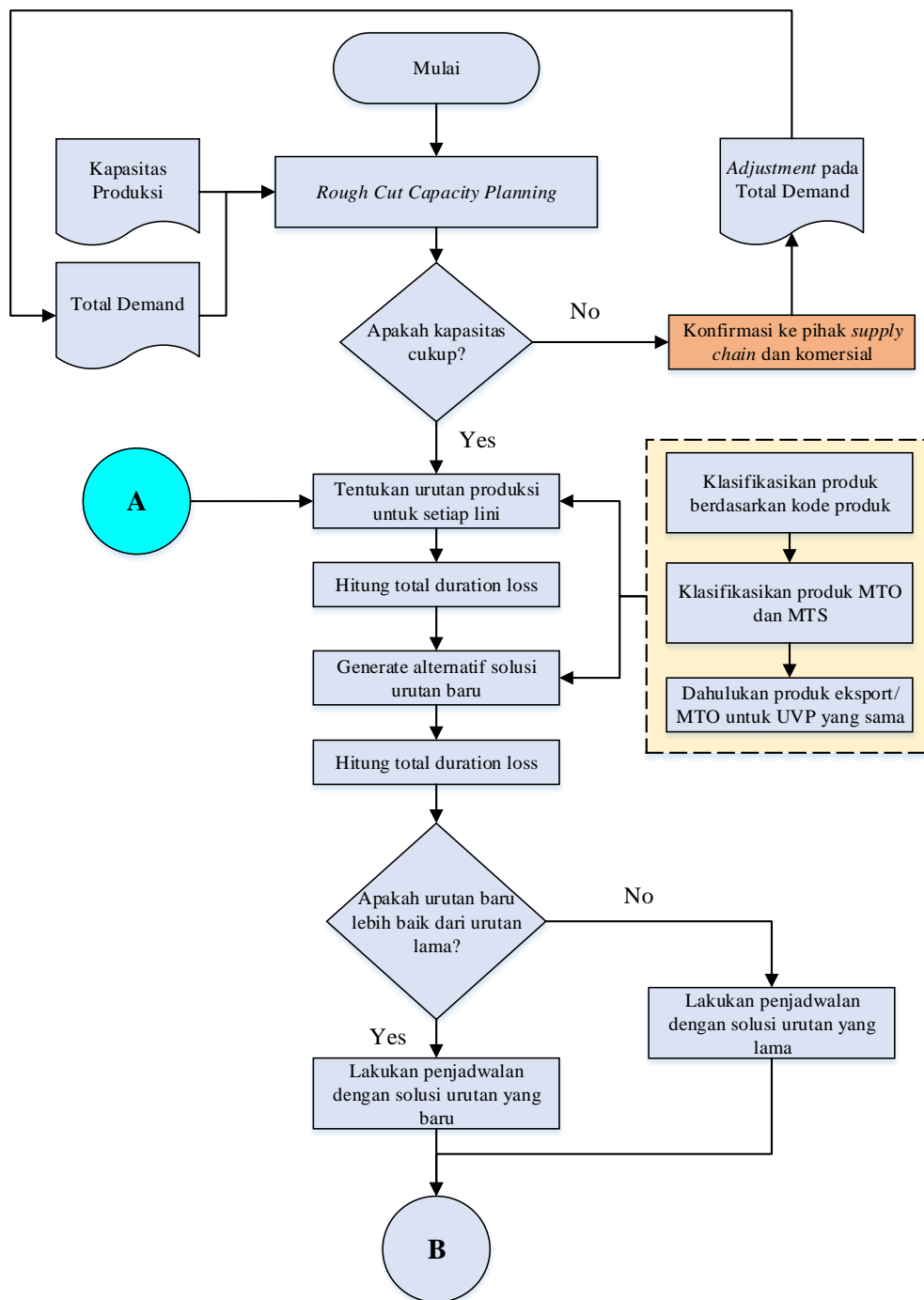


Gambar 3. 2 *Flowchart* Penentuan Lini Produksi

3.2.2 Penentuan Urutan Produk

Setelah menentukan lini produksi mana yang akan digunakan maka selanjutnya menentukan urutan produk yang telah dialokasikan ke dalam lini. Penentuan urutan produk ini mempertimbangan *dependent setup time* karena setiap kali ada perpindahan produk maka akan ada waktu *setup* yang muncul dan lamanya waktu *setup* tergantung dari produk apa yang diproduksi sebelumnya dan

apa yang hendak diproduksi. Penentuan urutan produk didasari oleh dua hal yaitu produk MTO (produk dengan kode selain ID) harus selalu dijadwalkan di awal. Kemudian produk dengan kode komersial produk yang sama selalu didekatkan. Gambar 3.3 menunjukkan alur dari penentuan urutan produk sebagai berikut:



Gambar 3. 3 Flowchart Penentuan Urutan Produk

3.2.3 Perhitungan *Makespan*

Selanjutnya akan dihitung berapa lama waktu untuk memproduksi produk yang telah dijadwalkan di depan (untuk produk-produk MTS). Dalam terminologi penjadwalan hal ini dapat disebut sebagai *makespan*. *Makespan* yang berarti lama waktu yang diperlukan untuk memenuhi keseluruhan permintaan produksi dalam satu bulan. Semua lini berjalan secara paralel sehingga lini dengan waktu penyelesaian yang paling panjang akan menjadi hasil dari *makespan*.

3.3 Tahap Validasi Model Penjadwalan Sistem Aktual

Validasi merupakan proses pengecekan apakah model yang dibuat telah sama dengan keadaan *real* sistem. Maka dari itu validasi dilakukan dengan cara membandingkan model penjadwalan sistem eksisting dengan data produksi bulan 2018 dan *planning board* tahun 2018. Selain itu, penulis juga melakukan validasi dengan cara melakukan wawancara langsung dengan *production planner* terkait model konseptual yang dibuat. Dari data *planning board* dapat dilihat bagaimana *production planner* menentukan lini dan urutan produk yang akan diproduksi. Validasi dilakukan dengan mengambil sebagian data pada bulan Maret 2018 khusus untuk produk *powder*. Tabel 3.1 menunjukkan data produksi pada bulan Maret 2018 sebagai berikut:

Tabel 3. 1 Permintaan Produksi Bulan Maret 2018

Kode Produk	Ukuran	Permintaan Maret (kg)
Al	5 kg	30720
A - ID	1 kg	163040
A - MY	1 kg	50000
A - PH	1 kg	31600
A - PK	1 kg	45000
A - ID	250 gr	39340
C - ID	100 gr	18080
F - ID	50 gr	4400
F - ID	250 gr	2970
IWP - BD	100 gr	1500
IWP - BD	50 gr	1200
L - ID	100 gr	7960
L - ID	15 gr	5958
T - ID	50 gr	3960

Kemudian dalam *flowchart* yang dibuat dilihat bahwa pertama kali yang dilakukan adalah penentuan lini mana yang akan digunakan untuk memproduksi produk. Maka kemudian dibangkitkan opsi lini yang memungkinkan untuk proses produksi sesuai dengan produk yang diinginkan. Berikut adalah opsi lini yang dibangkitkan sesuai dengan aturan yang ada dalam *flowchart*:

Tabel 3. 2 Opsi Lini tiap Kode Produk

Kode Produk	Ukuran	Opsi Lini			
		Lini 1	Lini 2	Lini 3	Lini 4
Al	5 kg	1	0	0	0
A - ID	1 kg	0	1	1	1
A - MY	1 kg	0	1	1	1
A - PH	1 kg	0	1	1	1
A - PK	1 kg	0	1	1	1
A - ID	250 gr	0	1	1	1
C - ID	100 gr	0	0	0	1
F - ID	50 gr	0	0	0	1
F - ID	250 gr	0	1	1	1
IWP - BD	100 gr	0	0	0	1
IWP - BD	50 gr	0	0	0	1
L - ID	100 gr	0	0	0	1
L - ID	15 gr	0	0	0	1

Angka biner 1 dan 0 pada Tabel 3.2 bisa tidaknya lini tersebut untuk memproduksi produk dengan ukuran yang diinginkan. Angka 1 menandakan produk tersebut mampu diproduksi dan angka 0 menyatakan sebaliknya. Dari kolom realisasi lini dapat dilihat bahwa lini yang terpilih sesuai dengan opsi lini yang dibangkitkan. Pada sistem asli apabila terdapat opsi lebih dari 1, lini akan dipilih dengan mempertimbangkan kesamaan jenis kode produk dan kapasitas. Contohnya apabila bahwa produk kode A hampir semua diproduksi di lini 2. Hal ini dilakukan agar tidak terjadi *setup* karena pergantian produk. Dalam algoritma pertimbangan ini juga sudah tercantum.

Tahap selanjutnya adalah melakukan validasi model konseptual terhadap sistem nyata mengenai urutan produk yang diproduksi dalam sebuah lini. Hal ini dilakukan dengan menggunakan data dari *planning board* PT Bayer Indonesia Divisi *Cropscience*. Dalam *planning board* terdapat informasi mengenai urutan produk yang diproduksi dalam sebuah lini, produk apa yang hendak diproduksi, berapa banyak yang harus diproduksi dan berapa jumlah *shift* yang dibutuhkan.

Dari model konseptual, dapat dilihat bahwa penentuan urutan produk dilakukan dengan aturan sebagai berikut:

1. Produk dengan kode yang sama diusahakan selalu didekatkan
2. Produk MTO (dengan kode selain ID) harus selalu didahulukan dibanding produk berkode ID.

Dari *planning board* dapat dilihat susunan urutan produk pada tiap lini. Urutan produk dapat dilihat dari Tabel 3.4 di bawah ini:

Dedicated Lini 1	A1-1kg (ID)					
Sharejob	A-1kg (ID)	A-1kg (MY)	A-1kg (PH)	A-1kg (PK)	A-250gr (ID)	F-250gr (ID)
Dedicated Lini 4	C-100gr (ID)	F-50gr (ID)	IWP-100gr (BD)	IWP-50gr (BD)	L-15gr (ID)	L-100gr (ID)

Tabel 3. 3 Susunan Urutan Produk tiap Lini

Lini Filling	Urutan Best					
1	A1-1kg (ID)					
2	A-1kg (PH)	A-1kg (MY)	A-1kg (PK)	A-1kg (ID)		
3	F-250gr (ID)	A-250gr (ID)				
4	L-100gr (ID)	L-15gr (ID)	F-50gr (ID)	C-100gr (ID)	IWP-100gr (BD)	IWP-50gr (BD)

Berdasarkan Tabel 3.3 dapat dilihat bahwa penentuan model konseptual yang dibuat telah sesuai dengan hasil dari *planning board* dimana dapat dilihat bahwa produk dengan kode selain ID didahulukan dan produk dengan kode yang sama didekatkan.

3.4 Tahap Pengembangan Algoritma Fase *Constructive*

Dalam tahap ini akan dijelaskan mengenai algoritma yang dikembangkan untuk menyelesaikan permasalahan penjadwalan pada lini paralel *filling* dengan mempertimbangkan *sequence dependent setup time*, *release time*, dan *machine eligibility restriction*. Pengembangan algoritma ini akan didasarkan pada jurnal yang telah disebutkan dalam *literature review*. Hasil dari algoritma ini akan

menjadi input untuk fase *improvement*. Algoritma pada fase *constructive* ini adalah: *longest processing time rule*, *least flexible job* dan *Hungarian method* (untuk *uniform parallel machine*). Algoritma fase *constructive* dibagi menjadi dua yaitu algoritma untuk produk *powder* dan *liquid* yang akan dijelaskan pada BAB 4.

3.5 Tahap Pengembangan Algoritma Fase *Improvement*

Pada tahap ini akan dikembangkan algoritma untuk fase *improvement*. Algoritma yang digunakan adalah *Variable Neighborhood Search (VNS)*. Fase ini diperlukan karena hasil dari solusi awal di fase *constructive* masih buruk dan terjebak dalam *local optimum*. Meskipun penerapan VNS belum tentu menjamin dapat menghasilkan hasil yang *global optimum* tetapi penerapan VNS ini dapat memperbaiki solusi yang sudah ada. Algoritma fase *improvement* akan dijelaskan pada BAB 4.

3.6 Tahap Verifikasi dan Validasi Algoritma

Sebelum melanjutkan ke problem komputasi di tahap selanjutnya maka perlu dilakukan validasi terhadap algoritma yang telah dikembangkan. Proses validasi ini bertujuan untuk memastikan bahwa algoritma yang akan digunakan telah sesuai dengan keadaan yang sebenarnya. Selain itu juga dilakukan verifikasi yaitu untuk memastikan apakah algoritma yang dibuat telah sesuai dengan model penjadwalan eksisting. Dengan dilakukannya proses validasi maka hasil dari komputasi dalam tahap eksperimen dapat dikatakan sesuai dengan keadaan yang sesungguhnya. Dalam penelitian ini akan dilakukan dengan data *dummy* yang akan diproses dengan algoritma yang telah dibuat, kemudian hasilnya akan dibandingkan model konseptual penjadwalan sistem aktual.

3.7 Tahap Eksperimen

Setelah algoritma yang telah dilakukan telah valid maka dapat dilakukan eksperimen. Dalam eksperimen ini akan dimasukkan data mengenai *demand* dan data penunjang lainnya dengan tujuan untuk melihat bagaimana algoritma yang telah dibuat dapat menjawab permasalahan yang ada. Kemudian akan dilakukan uji sensitivitas seperti misalnya apa yang terjadi jika konstrain yang ada dihilangkan,

jika terjadi kerusakan pada mesin, dan sebagainya. Uji sensitivitas penting untuk dilakukan dengan tujuan untuk mengevaluasi seberapa *robust* model yang telah dibuat.

3.8 Tahap Komparasi *Total Duration Loss* dan *Penalty Overtime*

Pada tahap ini dilakukan perbandingan total *duration loss* (faktor dari availabilitas yang mempengaruhi nilai OEE) sistem *real* dengan total *duration loss* dari hasil pengembangan algoritma yang diterapkan. Komparasi ini bertujuan untuk mengetahui seberapa baik penerapan algoritma yang telah dibuat dalam menyelesaikan permasalahan. Selain itu juga dilakukan perbandingan hasil *penalty overtime* dari algoritma dengan sistem *real*. Harapannya dengan penerapan algoritma ini dapat membantu *production planner* menyusun jadwal operasi produksi dengan baik dan membuat OEE menjadi lebih baik.

3.9 Tahap Analisis dan Interpretasi

Pada tahap ini dilakukan analisis terkait hasil dari eksperimen yang telah dilakukan. Analisis dilakukan untuk mengetahui perfomansi dari penggunaan algoritma yang digunakan untuk menyelesaikan permasalahan penjadwalan operasi produksi. Hasil dari analisa ini menjadi bagian penting dalam penelitian ini. Selain itu juga akan di analisa hasil dari perbandingan yang telah dilakukan pada tahap sebelumnya.

3.10 Kesimpulan dan Saran

Setelah selesai melakukan tahap analisis dan interpretasi maka akan dilakukan penarikan kesimpulan terkait hasil penelitian ini. Kesimpulan akan menjawab tujuan yang telah ditentukan pada bab I. Selain itu juga akan diberikan saran yang dapat dijadikan rekomendasi untuk penelitian yang akan dilakukan ke depannya.

(Halaman ini sengaja dikosongkan)

BAB 4

PENGUMPULAN DATA DAN PERANCANGAN ALGORITMA

Bab ini akan dibahas mengenai deskripsi umum perusahaan, pengumpulan data dan perancangan algoritma yang akan digunakan. Selanjutnya akan dilakukan proses verifikasi dan validasi terhadap model yang dibuat.

4.1 Deskripsi Perusahaan

PT Bayer Indonesia merupakan perusahaan yang berfokus pada *life science* dengan tujuan meningkatkan kualitas hidup manusia dengan memproduksi produk yang berhubungan dengan kesehatan dan pertanian. PT Bayer Indonesia terdiri dari 3 divisi yaitu *pharmaceuticals*, *health care* dan *crops science*. Penelitian ini berfokus pada Divisi *Crops science* yang berlokasi di Jalan Rungkut Industri I/12, Surabaya. Produk-produk yang dihasilkan oleh PT Bayer Indonesia Divisi *crops science* dibagi dalam 3 kelompok yaitu insektisida, fungisida dan herbisida dengan wujud produk berupa *powder*, *liquid* dan pasta. Produk yang dihasilkan antara lain Basta, Decis, Confidor dan lain-lain dengan ukuran produk yang bermacam-macam. Selain itu 40% hasil produksi diekspor ke Malaysia, Taiwan, Thailand, Australia, Pakistan dan Filipina. PT Bayer Indonesia Divisi *Crops science* menerapkan AMLS dimana telah ada perencanaan ke depan dalam hal *brand* produk apa yang harus menjadi *top sales*, peningkatan jumlah produksi, persiapan terhadap *warehouse space* dan lain-lain. Hal ini diupayakan agar PT Bayer Indonesia Divisi *Crops science* mampu bertahan dan bersaing dengan kompetitor perusahaan penghasil pestisida terbesar seperti misalnya BASF, Arysta dan Syngent. Selain itu juga ada penerapan *operation excellence* yang terdiri dari 5S, TPM, TQM, *stability and capability*, *robustness*. Bayer memiliki 5 *core strengths* yang terdiri dari *innovation*, *customer centricity*, *quality*, *process excellence*, *portfolio management* dan *people*. Segala strategi dan kegiatan operasional yang dijalankan oleh PT Bayer Indonesia Divisi *Crops science* harus disesuaikan dengan misi dan nilai dari perusahaan yaitu “*Science for a Better Life*” disertai dengan *LIFE Values*.

4.2 Pengumpulan Data

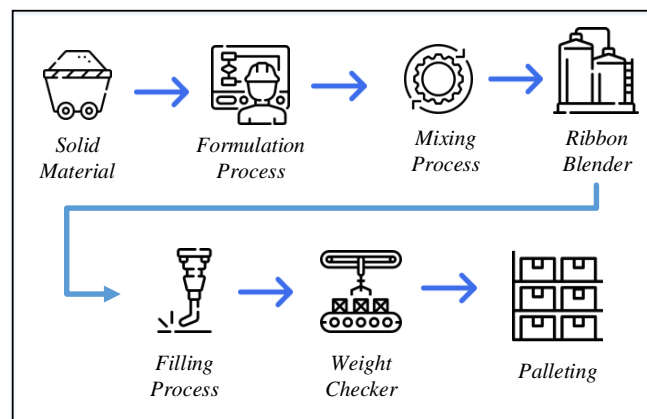
Dalam subbab ini akan dijelaskan mengenai data yang berhubungan dalam proses penjadwalan mulai dari pemahaman mengenai alur proses produksi hingga data-data *demand*, variasi produk dan sebagainya.

4.2.1 Alur Lini Produksi

Pabrik PT Bayer di Surabaya terdiri dari 9 lini proses *filling* yang terdiri dari 4 lini untuk memproduksi produk *liquid*, 4 lini untuk memproduksi produk *powder* dan 1 lini untuk memproduksi produk pasta. Penelitian ini membatasi hanya sampai produk *powder* sehingga produk pasta tidak ditampilkan. Proses pembuatan untuk masing-masing wujud produk tersebut berbeda-beda dan juga tergantung pada aktivitas produksi yang dilakukan untuk menghasilkan produk tersebut.

4.2.1.1 Produk Powder

Alur produksi produk wujud *powder* yang dimulai dari proses *raw material* kemudian dimasukkan ke dalam proses formulasi hingga akhirnya menjadi produk jadi dan siap untuk dikirim ditampilkan pada Gambar 4.1



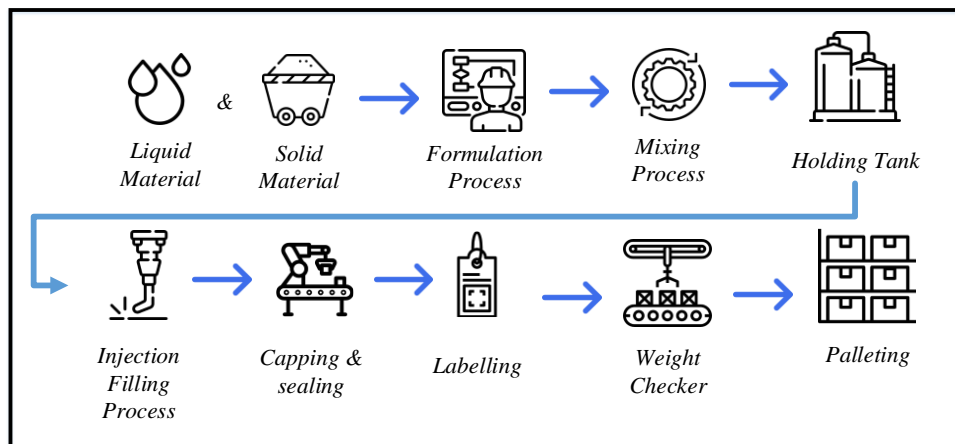
Gambar 4. 1 *Flow Process* Produk Wujud *Powder*

Gambar di atas merupakan alur proses produksi untuk aktivitas produksi formulasi kemudian dilanjutkan dengan *filling*. Apabila aktivitas produksi merupakan *refilling* maka akan langsung masuk ke dalam *ribbon blender* dan dilakukan *filling process*, pemeriksaan ukuran produk dan

palleting. Proses produksi tersebut dijalankan dalam lini yang terdiri dari 3 bagian utama yaitu *mixing tank*, *holding tank* yang berfungsi sebagai penyimpanan sementara hasil produk formulasi maupun produk *refilling* dan yang terakhir adalah mesin *filling*.

4.2.1.2 Produk Liquid

Dimulai dari memproses *raw material* yang terdiri dari *liquid material* dan *solid material*. Kemudian dimasukkan dilanjutkan proses formulasi hingga akhirnya menjadi produk jadi dan siap untuk dikirim seperti Gambar 4.3 di bawah ini:



Gambar 4. 2 Flow Process Produk Wujud Liquid

Apabila aktivitas produksi merupakan *refilling* maka produk setengah jadi akan langsung masuk ke dalam *holding tank* dan dilakukan *injection filling process*. Setelah itu maka akan ada pemberian tutup botol, *seal* dan label pada kemasan luar produk. Ukuran produk yang tidak sesuai akan masuk dalam *reject product* kemudian produk yang lolos standar akan ditata dalam *pallet* dan akan diperiksa kembali ukurannya.

Proses produksi tersebut dijalankan dalam lini-lini yang tersedia dimana dalam lini terdapat 3 bagian utama yaitu tanki formulasi, *holding tank* yang berfungsi sebagai penyimpanan sementara hasil produk formulasi maupun produk *refilling* dan yang terakhir adalah mesin *injection filling*.

4.2.2 Kapasitas Lini *Filling*

Kapasitas lini *filling* produksi menyesuaikan dengan ukuran produknya. Hal inilah yang kemudian membuat permasalahan lini *filling* produksi memiliki *constraint* berupa *machine eligibility restriction* yang berarti tidak semua lini *filling* dapat digunakan untuk memproduksi produk. Selain itu juga ada kapasitas maksimal lini *filling* untuk produk *powder* ukuran 15-gram hingga 25-kilogram dan produk *liquid* mulai dari ukuran 50-ml hingga 200.000-ml.

4.2.4 Matriks *Acceptance Contamination Limit* (ACL)

ACL merupakan hal penting yang harus diperhatikan dalam proses penjadwalan. Sebagai pabrik pestisida, tentunya penting untuk menjaga agar produk yang dihasilkan tidak terkontaminasi. Salah satu pemicu terjadinya kontaminasi adalah produk dengan *active ingredients* yang berbeda. Maka dari itu PT Bayer telah menetapkan standar mengenai matriks ACL dimana di dalam matriks tersebut berisi batasan ppm yang diperbolehkan antara produk satu dengan produk lainnya. Batasan ppm ACL ini menunjukkan kadar batasan produk (i-1) yang masih boleh tersisa di dalam mesin yang hendak digunakan untuk memproduksi produk (i). Pengecekan ini dilakukan setiap kali dilakukan proses *setup*. Dalam matriks juga ditunjukkan bahwa terdapat larangan suatu produk (i) diproduksi setelah produk (i-1).

4.2.5 Jenis Produk dan Variasi Ukuran Produk

Dalam PT Bayer Indonesia Divisi *Cropscience* jenis produk disebut sebagai UVP atau dengan kata lain merupakan induk dari produk. Kemudian dari tiap UVP memiliki nama komersial produk dengan variasinya masing-masing. Tabel 4.4 dan Tabel 4.5 berisi nama komersial (kode produk) dan variasi produk yang ada dalam PT Bayer Indonesia Divisi *Cropscience* mulai dari wujud *powder* hingga *liquid* dengan total 89.

Tabel 4. 1 Variasi Produk Wujud *Powder*

Kode	Variasi (dalam gr)									Total
	15	50	100	250	500	1k	5k	10k	25k	
T		T-50								17
Al				Al-250			Al-5k	Al-1k		
C			C-100							
IWP		IWP - 50	IWP - 100		IWP - 500					
A				A-250	A-500	A-1k			A-25k	
F		F-50	F-100	F-250						
L	L-15		L-100							

4.2.6 Setup Time

Terdapat 3 jenis proses *setup* yang ada yaitu ketika mengganti produk produk tanpa pergantian ukuran, pergantian ukuran tanpa pergantian produk dan pergantian produk disertai pergantian ukuran. Contohnya adalah pergantian dari F-50 menuju ke T-50. Tahapan diawali dari pengosongan mesin dari produk-produk (i-1) kemudian mesin akan dibersihkan dengan menggunakan bahan yang tepat seperti misalnya dalam produk *powder* menggunakan kaolin. Kemudian akan diambil sampel untuk dibawa ke QA untuk dicek kadar ACL-nya.

Yang kedua adalah pergantian ukuran tanpa pergantian produk. Proses *setup* ini merupakan proses yang paling singkat dibanding 2 jenis *setup* lainnya karena hanya melakukan pergantian ukuran *nozzle* pada mesin *filling*. Contohnya adalah pergantian dari F-50 menuju ke F-250

Proses *setup* berikutnya merupakan gabungan dimana terjadi proses pergantian produk disertai pergantian ukuran. Seperti misalnya F-50 menjadi L-15. Data mengenai matriks *setup time* keseluruhan akan dicantumkan dalam lampiran. Contoh matriks *setup time* yang digunakan ditampilkan pada Tabel 4.6

Tabel 4. 2 Contoh Matriks *Setup Time* Produk *Powder*

			1	2	3	4	5	...	26	27
			T-50	Al-250	Al-250	Al - 5	Al - 5	...	L-100	L-100
			ID	ID	MY	ID	PH	...	ID	BD
1	T-50	ID	0	3.37	3.37	2.72	2.72	...	3.07	3.07
2	Al-250	ID	2.18	0	0	1.38	1.38	...	1.77	1.77
3	Al-250	MY	2.18	0	0	1.38	1.38	...	1.77	1.77
4	Al - 5	ID	3.25	1.12	1.12	0	0	...	1.12	1.12
5	Al - 5	PH	3.25	1.12	1.12	0	0	...	1.12	1.12
...
25	L-15	ID	2.12	3.48	3.48	2.62	2.62	...	1.17	1.17
26	L-100	ID	2.58	3.00	3.00	1.30	1.30	...	0	0
27	L-100	BD	2.58	3.00	3.00	1.30	1.30	...	0	0

4.2.7 Data Permintaan Produk dan *Processing Time*

Processing time merupakan lamanya waktu yang diperlukan untuk menyelesaikan suatu proses produksi. *Processing time* (dalam satuan jam) didapatkan dengan membagi *demand* dengan kapasitas produksi mesin *filling*.

Berikut adalah data permintaan produksi dan *processing time* dari produk *liquid* dimana lini produksi 1 dan 3 memiliki kecepatan yang sama sedangkan lini 2 memiliki kecepatan yang lebih rendah. Data *processing time* produk *liquid* secara menyeluruh akan dicantumkan dalam lampiran, berikut adalah contoh data permintaan produksi dan *processing time* dari produk *liquid* bulan Maret:

Tabel 4. 3 Contoh *Demand* Produksi Produk *Liquid*

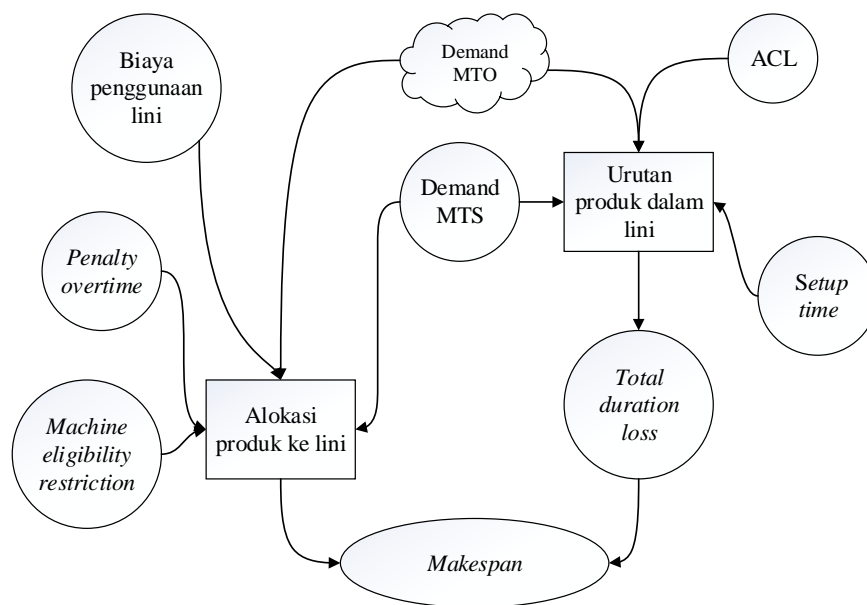
Produk ke-	Kode Produk	Kode Negara	Demand	Kapasitas di lini 1/3	Kapasitas di lini 2	PT Lini 1/3	PT Lini 2
2	Ag - 1000 mL	ID	14400	16700	-	7	999
5	Bc - 100 mL	ID	5600	3700	2300	14	21
6	Bc - 500	ID	55650	10400	7500	42	56
7	Bf - 250 mL	ID	7920	6000	4800	14	14
...
67	HSL - 200 L	PH	8000	16000	-	7	999
68	IMD - 200 L	ID	4800	16000	-	7	999
75	Mov - 100 mL	ID	2930	3700	2300	7	14
81	PSL - 100 mL	ID	4000	3700	2300	14	14
83	PSL - 500 mL	ID	21390	10400	7500	21	21

Berdasarkan Tabel 4.8 dapat dilihat bahwa produk 2 memiliki *processing time* di lini 2 sebesar 999. Pemberian nilai 999 ini bertujuan agar model tidak mengalokasikan produk 2 ke lini 2 melainkan ke lini lain yang memiliki *processing time* yang lebih rendah.

4.3 Perancangan Algoritma

Perancangan algoritma ini digunakan untuk menyelesaikan permasalahan penjadwalan yang ada pada PT Bayer Indonesia Divisi *Cropscience*. Perancangan algoritma ini dilakukan secara bertahap sehingga menjawab semua *constraint* yang ada dan juga mencapai tujuan yang diinginkan. Algoritma yang dikembangkan didasarkan pada jurnal-jurnal yang telah dijelaskan pada subbab *literature review*. Sebelum merancang algoritma untuk menyelesaikan suatu permasalahan, perlu dibuat model permasalahannya terlebih dahulu. Model dapat dibuat dengan

mengembangkan model matematika ataupun dengan model konseptual. Model konseptual yang digunakan adalah *influence diagram* dimana bentuk oval menunjukkan *output*, lingkaran menunjukkan *system variable* atau *controllable input*, kotak menunjukkan *decision variable* dan awan menunjukkan *input variable* yang tidak dapat dikontrol. Selain itu juga terdapat anak panah yang menunjukkan *influence relationship*. Gambar di bawah ini merupakan model konseptual permasalahan penjadwalan produksi yang ada di PT Bayer Indonesia Divisi *Cropscience* dengan menggunakan *influence diagram*:



Gambar 4. 3 *Influence Diagram* Permasalahan Penjadwalan

Berdasarkan Gambar 4.5 dapat dilihat bahwa tujuan utama adalah meminimasi *makespan* yang dipengaruhi oleh *total duration loss*. *Decision variable* ada dua yaitu alokasi produk ke lini dan urutan produk dalam lini. Alokasi produk dalam lini dipengaruhi oleh biaya penggunaan lini, *penalty overtime*, *machine eligibility restriction* dan permintaan produksi MTO-MTS. Sedangkan urutan produk dalam lini dipengaruhi oleh *setup time*, *ACL* (*Acceptance Contamination Limit*), *setup time*, dan permintaan produksi MTO-MTS. Kemudian dari model konseptual inilah dikembangkan perancangan algoritma yang dapat menyelesaikan permasalahan penjadwalan yang ada.

Perancangan algoritma dibagi menjadi dua yaitu fase *constructive* yang bertujuan untuk membangun *initial solution* kemudian fase *improvement* dengan *Variable Neighborhood Search* (VNS). Pendekatan LFJ-LPT dalam fase *constructive* diterapkan untuk meminimasi *makespan* pada penjadwalan lini paralel yang bersifat identik. Selain itu, *hungarian method* juga diterapkan untuk menentukan alokasi produk dalam lini untuk lini paralel yang bersifat *uniform*. Hasil yang memungkinkan terjebak di lokal optimal sehingga diperlukannya pencarian struktur tetangga dengan VNS yang dapat memberikan hasil yang lebih baik.

Algoritma dibagi menjadi dua yaitu untuk produk *powder* dan produk *liquid*. Perbedaan utama penjadwalan kedua produk ini ada pada *machine configuration*-nya yang mana pada produk *powder* bersifat *identic parallel machine* sedangkan pada produk *liquid* bersifat *uniform parallel machine*. Terminologi *identic parallel machine* digunakan ketika proses penjadwalan dilakukan pada lebih dari 1 mesin dengan kecepatan produksi yang sama. Sedangkan terminologi *uniform parallel machine* digunakan ketika proses penjadwalan dilakukan pada lebih dari 1 mesin dengan kecepatan produksi yang berbeda.

Tujuan utama dari penjadwalan ini adalah meminimasi *makespan* dengan pertimbangan *sequence dependent setup times*, *machine eligibility restriction* dan *release time*. Penjadwalan mempertimbangkan *sequence dependent setup times* karena urutan penjadwalan produk mempengaruhi lamanya proses setup sehingga akan ada matriks *setup time* yang menunjukkan lamanya waktu perubahan dari satu produk ke produk lainnya. Penjadwalan yang dibuat akan berusaha untuk meminimasi total waktu *setup* yang diwakilkan dengan meminimasi *makepsan*.

Setiap lini memiliki batasan kapasitas waktu produksi yaitu 420 jam dalam 1 bulan yang mana didapat dari perhitungan 20 hari kerja x 3 shift x 7 jam. Apabila saat penjadwalan awal menghasilkan *completion time* tiap lini melebihi 420 jam maka akan diaktifkan kerja *overtime* yang memberikan *penalty cost*. Penjadwalan yang dibuat berusaha untuk meminimasi *penalty cost* yang diwakilkan dengan meminimasi *makespan*.

Release time dipertimbangkan karena tidak semua produk diketahui saat $t=0$ yaitu untuk produk MTO. Dimana *processing time* dari produk MTO baru diketahui

saat produk datang pada *release time* tertentu. Sedangkan saat $t=0$ yang dapat dilakukan adalah melakukan proses penjadwalan untuk produk-produk MTS. Ketika produk MTO datang maka akan dicek lini mana yang mampu memproduksi produk tersebut kemudian akan diselipkan produk MTO pada lini tersebut. Proses memasukan produk MTO ke dalam urutan penjadwalan dilakukan dengan tidak menginterupsi proses produksi yang sedang berjalan melainkan menunggu hingga *lot* produksi untuk produk tersebut selesai. Setelah itu akan ditentukan kembali urutan penjadwalan dari produk-produk sisa yang belum diproduksi dengan mempertimbangkan kehadiran produk MTO yang ada sekarang.

4.3.1 Perancangan Algoritma Produk *Powder*

Perancangan algoritma produk *powder* didasari dengan menggunakan algoritma *least flexible job* dan *longest processing time*. Pada produk *powder* terdapat 4 lini *filling* dimana terdapat produk yang hanya bisa diproduksi pada lini tertentu saja. Contohnya ukuran produk mulai dari 4 kg hingga 25 kg hanya dapat diproduksi pada lini *filling* 1. Selain itu ukuran produk kecil seperti 15 gram hingga 100 gram hanya bisa diproduksi di lini *filling* 4. Dalam algoritma *least flexible job*, *job* akan diurutkan berdasarkan pilihan mesin yang paling sedikit hingga *job* yang memiliki pilihan mesin yang banyak. *Least flexible job* akan mendahulukan untuk menjadwalkan produk dengan jumlah mesin yang paling sedikit ($M = 1$). Setelah itu akan dikumpulkan *job-job* atau produk-produk yang hanya dapat diproduksi di lini *filling* 1 dan lini *filling* 4 dengan nama set *J*. Kemudian akan dibangkitkan seluruh kemungkinan urutan produk. Urutan produk yang memberikan total setup time terendah akan dipilih.

Sedangkan untuk produk yang memiliki pilihan lini lebih dari 1 ($M_j > 1$), penentuan alokasi produk pada lini didasarkan oleh pertimbangan *longest processing time*. Algoritma LPT pada mesin paralel dengan tujuan minimasi *makespan* berguna untuk mencegah ketidakseimbangan antara mesin yang satu dengan mesin yang lainnya. *Job* dengan *processing time* terpanjang akan dialokasikan terlebih dahulu kemudian *job* dengan *processing time* yang pendek akan berguna untuk menyeimbangkan muatan pada mesin-mesin yang ada.

Selanjutnya untuk memperbaiki hasil dari *initial solution* $M_j > 1$ maka hasil urutan di lini-lini tersebut akan diperbaiki dengan masuk ke dalam tahap *improving solution*. Tahap ini dilakukan dengan menggunakan *Variable Neighborhood Search* dengan input berupa: output tahap 3 yaitu urutanbest, *makespan* dan *total cost* serta *processing time* dan *matriks setup time*. Penjelasan mengenai VNS telah ada pada subbab *literature review* dimana di sini antar lini dilakukan dengan menggunakan VNS-job series exchange sedangkan kalau di dalam lini dilakukan dengan menggunakan VNS-job insertion dan VNS-job exchange. Berikut adalah algoritma dari penjadwalan produk *powder* saat $t=0$ (untuk produk MTS) yang terdiri dari tahap 1 (*initial solution*) dan tahap 2 (*improving solution*):

Tabel 4. 4 Algoritma Tahap 1- *Generating Initial Solution* Produk Powder

<i>Tahap 1: Generating Initial Solution</i>	
<p>Input: matriks produk-mesin, processing time, matriks setup time, <i>penalty overtime cost</i> Output: makespan, completion time tiap lini, total setup time dan total penalty overtime cost</p>	
<p>Step 1: Least Flexible Job Algorithm Set M_j = jumlah lini yang mampu memproses produk j Urutkan produk dari M_j terkecil hingga M_j terbesar Gabungkan produk dengan $M_j = 1$</p>	
<p>Step 2: Untuk $M_j = 1$ Set J sebagai job yang mampu diproduksi di lini M_i Bangkitkan semua kemungkinan urutan job di set J Pilih urutan yang memberikan total setup time terendah Completion time = processing time + total setup time If completion time > 420 Penalty cost = (total waktu overtime * pinalti overtime per jam) end Set urutanbest sebagai urutan job dengan total setup time terendah Output = [Urutanbest, penalty cost, completion time, total setup time] Selesai</p>	
<p>Step 3: Untuk $M_j > 1$ Longest Processing Time Algorithm Set k = urutan job berdasarkan processing time terpanjang Masukan job urutan $k=1$ ke lini M_i, hitung completion time Masukan job urutan $k+1$ ke lini M_{i+1}, hitung completion time If comp time lini $M_i > M_{i+1}$ Job urutan $k+1$ masuk ke lini M_{i+1} Else Job urutan $k+1$ masuk ke lini M_i End Ulangi hingga seluruh job telah masuk dalam lini M_i Untuk setiap M_i, completion time = processing time + total setup time If completion time > 420 Penalty cost = (total waktu overtime * pinalti overtime per jam) end Set urutanbest sebagai urutan job pada tiap lini M_i Makespan = Max (Completion time lini M_i, M_{i+1}, \dots) Output = [Urutanbest, penalty cost, makespan, total setup time] Selesai Masuk tahap improving solution</p>	

Tabel 4. 5 Algoritma Tahap 2: *Improving Solution* - VNS

<i>Tahap 2: Improving Solution – VNS Algorithm</i>
<p>Input: urutanbest, makespan, processing time dan matriks setup time</p> <p>Output: makespan, completion time tiap lini, total setup time dan total penalty overtime cost</p> <p>Untuk $M_j > 1$</p> <p>Step 1:</p> <p style="padding-left: 40px;">Set $T = \text{Makespan}$</p> <p style="padding-left: 40px;">Set maxiter</p> <p style="padding-left: 40px;">Set alpha (faktor penurunan suhu)</p> <p>Step 2:</p> <p style="padding-left: 40px;">While $T > 1e-8$</p> <p style="padding-left: 80px;">Lakukan VNS - job series exchange</p> <p style="padding-left: 80px;">Hasil : makespanbest_new</p> <p style="padding-left: 80px;">If makespanbest_new < makespan</p> <p style="padding-left: 120px;">Urutanbest = urutanbest hasil job series exchange</p> <p style="padding-left: 80px;">Else</p> <p style="padding-left: 120px;">Delta_makespan = makespanbestnew – makespanbest</p> <p style="padding-left: 120px;">PE = $\exp(-\text{delta_makespan}/T)$</p> <p style="padding-left: 120px;">r = bilangan random</p> <p style="padding-left: 120px;">If $r < PE$</p> <p style="padding-left: 160px;">Urutanbest = urutan hasil job series exchange</p> <p style="padding-left: 120px;">End</p> <p style="padding-left: 80px;">End</p> <p style="padding-left: 80px;">Iterasi = iterasi + 1;</p> <p style="padding-left: 80px;">If iterasi > Maxiter</p> <p style="padding-left: 120px;">$T = T * \alpha$</p> <p style="padding-left: 120px;">Iterasi kembali ke 1</p> <p style="padding-left: 80px;">End</p> <p>End</p> <p>Output = [Urutanbest, makespan, total setuptime]</p> <p>Step 3:</p> <p>Untuk setiap lini dengan urutanbest hasil job series exchange:</p> <p>Lakukan VNS - job insertion atau VNS – job exchange</p> <p>Hasil : makespanbest_new2</p> <p style="padding-left: 40px;">If makespanbest_new2 < makespanbest_new</p> <p style="padding-left: 80px;">Urutanbest = urutanbest hasil job insertion / job exchange</p> <p style="padding-left: 40px;">Else</p> <p style="padding-left: 80px;">Urutanbest = urutanbest hasil step 2</p> <p style="padding-left: 40px;">End</p> <p>Output = [Urutanbest, makespan, total setuptime]</p>

4.3.2 Perancangan Algoritma Produk *Liquid*

Penjadwalan terhadap produk yang memiliki $M_j = 1$ dan $M_j > 1$ dengan kecepatan mesin yang identik untuk produk *liquid* sama seperti algoritma produk *powder*. Namun yang berbeda di sini adalah untuk $M_j > 1$ dengan kecepatan mesin yang tidak sama (*uniform*). Lini 2 merupakan lini yang memiliki kecepatan lebih rendah dibandingkan lini 1 dan lini 3 dalam memproduksi produk dengan ukuran 50 ml hingga 500 ml. Produk dengan permintaan tinggi harus dialokasikan pada lini yang memiliki kecepatan tinggi sehingga *processing time* dapat lebih cepat. Chuang, et al. (2010) menggunakan *Hungarian Method* untuk menyelesaikan problem *assignment* ini dengan tujuan meminimasi total *completion time*. Berikut adalah model matematis *assignment problem* yang digunakan dalam penelitian ini sebagai tahap *initial solution* pada mesin yang memiliki kecepatan berbeda:

Objective Function:

$$\text{Minimize } Z = \sum_{j=1}^j \sum_{i=1}^i Pt_{ij} * X_{ij} \quad (4.1)$$

Subject To:

$$\sum_{j=1}^j X_{ij} = 1 \quad \text{for } i=1 \dots i \quad (4.2)$$

$$\sum_{i=1}^i X_{ij} \geq 1 \quad \text{for } j=1 \dots j \quad (4.3)$$

$$\sum_{i=1}^i X_{ij} \times pt_{ij} \leq kap_j \quad \text{for } j=1 \dots j \quad (4.4)$$

$$X_{ij} \in \text{biner} \quad \text{for } i=1 \dots i, \text{ for } j=1 \dots j \quad (4.5)$$

Persamaan (4.1) menunjukkan *objective function* yang berusaha meminimumkan *completion time* dengan menjumlahkan hasil perkalian *processing time* dengan keputusan mengalokasikan produk *i* pada lini *filling j*. Persamaan (4.2) memastikan bahwa setiap produk hanya diproduksi pada satu mesin saja. Persamaan (4.3) menunjukkan bahwa lini *filling j* dapat memproses lebih dari satu produk. Persamaan (4.4) menunjukkan bahwa pengalokasikan produk pada lini

filling j tidak boleh melebihi kapasitas lini *filling j*. Berikut adalah algoritma dari penjadwalan produk *liquid* saat $t=0$ (untuk produk MTS) yang terdiri dari tahap 1 (*generating initial solution*) dan tahap 2 (modifikasi VNS-*job series exchange*):

Tabel 4. 6 Algoritma Tahap 1: *Generating Initial Solution* Produk *Liquid*

Tahap 1: Generating Initial Solution	
Input:	matriks produk-mesin, processing time, matriks setup time, <i>penalty overtime cost</i>
Output:	makespan, completion time tiap lini, total setup time dan total <i>penalty overtime cost</i>
Step 1: Least Flexible Job Algorithm	
	Set M_j = jumlah lini yang mampu memproses produk j
	Urutkan produk dari M_j terkecil hingga M_j terbesar
	Gabungkan produk dengan $M_j = 1$
Step 2:	
	Untuk $M_j = 1$
	Set J sebagai job yang mampu diproduksi di lini M_i
	Bangkitkan semua kemungkinan urutan job di set J
	Pilih urutan yang memberikan total setup time terendah
	Completion time = processing time + total setup time
	If completion time > 420
	Penalty cost = (total waktu overtime * <i>pinalti overtime per jam</i>)
	end
	Set urutanbest sebagai urutan job dengan total setup time terendah
	Output = [Urutanbest, penalty cost, completion time, total setup time]
	Selesai
Step 3:	
	Untuk $M_j > 1$ – kecepatan mesin berbeda
	Lakukan Hungarian Method untuk menentukan job diproduksi di lini mana
	Completion time = processing time + total setup time
	If completion time > 420
	Penalty cost = (total waktu overtime * <i>pinalti overtime per jam</i>)
	end
	Set urutanbest sebagai alokasi job pada lini hasil Hungarian Method
	Output = [Urutanbest, penalty cost, completion time, total setup time]
	Selesai
	Untuk $M_j > 1$ – kecepatan mesin sama
	Tahapan sama seperti Tahap 1 – Step 3 Algoritma Powder
	Masuk tahap improving solution

Perbedaan penerapan VNS-*job series exchange* pada algoritma produk *liquid* adalah terdapat produk yang tidak boleh terlibat dalam VNS-*job series exchange*. Lini *filling* 3 merupakan lini *filling dedicated* (untuk produk ukuran 5000 ml) sekaligus lini *filling* yang bisa *sharing* dengan lini lain (untuk produk ukuran 50 ml – 1000 ml). Hal inilah yang kemudian menyebabkan dalam proses VNS-*job series exchange* perlu ditambahkan pengecekan apakah produk ukuran 5000 ml tetap berada pada lini 3 atau berpindah karena proses VNS-*job series exchange*. Ketika ikut berpindah maka akan urutan baru yang dihasilkan akan dibuang dan dikembalikan pada urutan yang sebelumnya serta akan menuju ke iterasi berikutnya.

4.3.3 Perancangan Algoritma Produk MTO *Powder* dan *Liquid*

Kedatangan produk MTO yang tidak menentu menyebabkan penjadwalan yang telah dibuat perlu diperbaiki dengan mendahulukan produk MTO untuk diproduksi. Saat $t=0$ perusahaan hanya dapat melakukan penjadwalan produksi untuk produk-produk MTS dan berdasarkan hasil wawancara dengan pihak PT Bayer Indonesia Divisi *Cropscience* disebutkan bahwa kedatangan produk MTO bisa terjadi 5 hingga 10 kali sehingga penjadwalan yang telah dibuat saat $t=0$ harus diubah. Notasi yang digunakan adalah:

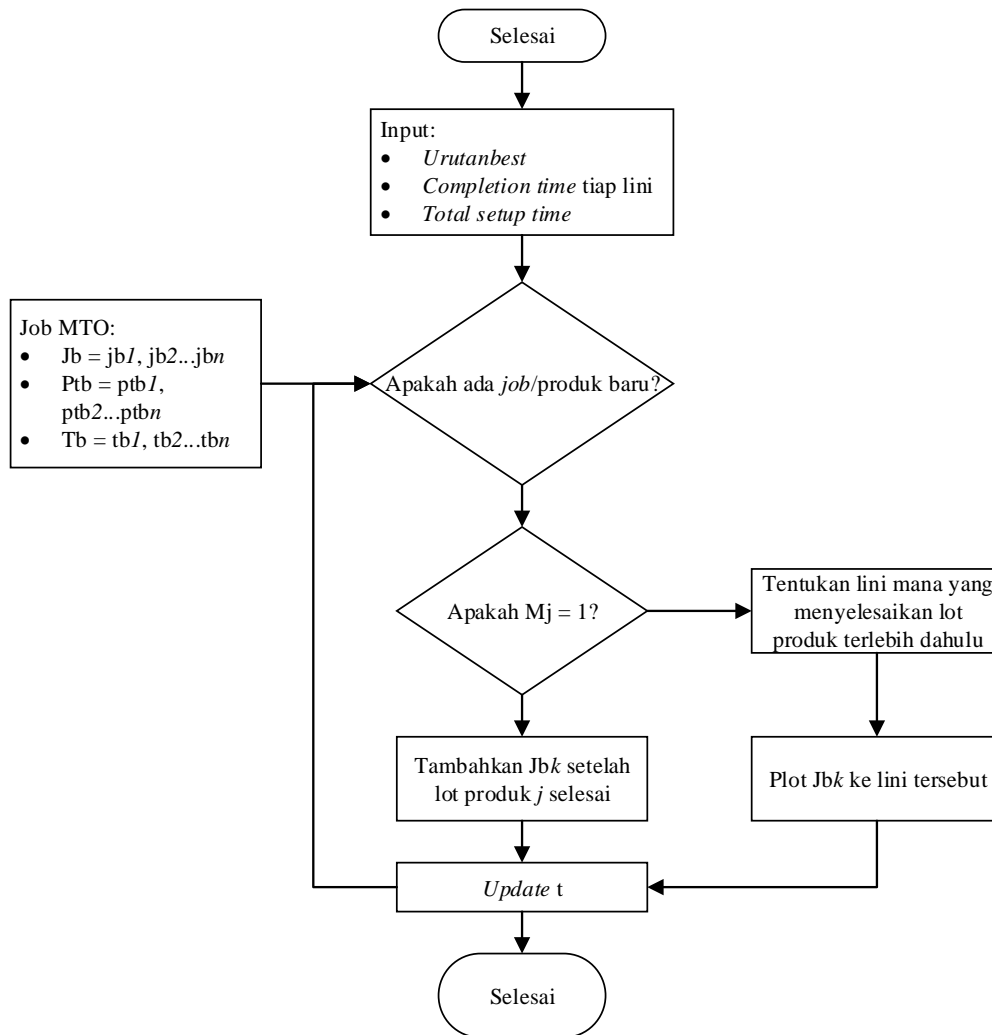
Jb_k : Kode produk *job baru* yang terdiri dari Jb_1 hingga Jb_n

Ptb_k : *Processing time* dari tiap Jb_k

Tb_k : Kedatangan Jb_k saat t

Setiap kali ada produk baru yang datang maka perlu dicek apakah produk tersebut hanya dapat diproduksi di 1 lini ($M_j=1$) atau lebih. Apabila $M_j=1$ maka produk langsung dimasukkan ke dalam penjadwalan dengan cara memotong penjadwalan yang selanjutnya dan menyelipkan produk baru setelah *lot* produksi yang sedang dikerjakan sekarang telah selesai.

Sedangkan untuk produk baru Jb yang dapat diproses lebih dari 1 lini ($M_j > 1$) maka akan dilihat lini mana yang terlebih dahulu menyelesaikan *lot* produksinya. Lini yang lebih dahulu menyelesaikan akan menjadi lini terpilih. Gambar 4.6 menunjukkan tahapan memasukan produk baru MTO ke dalam penjadwalan:



Gambar 4. 4 Flowchart Penjadwalan dengan Produk MTO

4.4 Verifikasi dan Validasi

Proses verifikasi dan validasi penting untuk dilakukan dengan tujuan untuk mengecek apakah algoritma yang dibuat telah sesuai atau tidak dan mampu menghasilkan *output* yang diharapkan oleh perusahaan. Proses verifikasi dan validasi juga dilakukan dengan cara melakukan diskusi dengan pihak *production planner*. Proses verifikasi dan validasi ini dilakukan dengan cara membandingkan hasil komputasi dari algoritma dengan hasil dari perhitungan manual (enumerasi). Algoritma yang telah dibuat di-*running* dengan menggunakan *software* Matlab R2015a. Percobaan enumerasi ini dilakukan dengan data *dummy*. Verifikasi dan validasi dilakukan terhadap algoritma produk *powder* dan *liquid*.

4.4.1 Verifikasi dan Validasi Algoritma *Powder*

Proses enumerasi dilakukan menggunakan data *dummy* di bawah ini:

Terdapat 10 produk MTS dan 3 produk MTO dengan *release date* yang berbeda-beda setiap produknya. Terdapat 4 lini dengan kecepatan proses produksi yang sama. Biaya yang dikeluarkan apabila menggunakan lini adalah Rp 10.000.000 dan *penalty overtime cost* adalah Rp 35.000 per jam yang akan dikenakan jika total *completion time* dalam suatu lini tertentu melebihi 420 jam.

Tabel 4. 7 Permintaan Produksi Produk MTS

Permintaan Produksi Saat $t=0$					
<i>Job</i>	Ukuran Produk	Kg/ jam	Permintaan Produksi (kg)	<i>Processing time</i> (dalam jam)	Opsi Lini
1	5 kg	285.71	48000	168.00	1
2	10 kg	428.57	21000	49.00	1
3	25 kg	714.29	60000	84.00	1
4	250 gr	485.71	150000	308.82	2,3,4
5	250 gr	485.71	45000	92.65	2,3,4
6	500 gr	800.00	54000	67.50	2,3,4
7	500 gr	800.00	36000	45.00	2,3,4
8	1 kg	1171.43	60000	51.22	2,3,4
9	15 gram	42.86	7500	175.00	4
10	50 gram	114.29	6000	52.50	4

Berdasarkan Tabel 4.12 di atas maka kemudian dilakukan pengelompokan terhadap produk-produk yang hanya memiliki 1 opsi saja. Berikut adalah hasilnya:

Tabel 4. 8 Pengelompokan *Job* ke dalam Lini

<i>Dedicated Lini 1</i>	1	2	3		
<i>Sharejob</i>	4	5	6	7	8
<i>Dedicated Lini 4</i>	9	10			

Setelah itu dilakukan penjadwalan terhadap lini yang *dedicated* yaitu lini 1 dan lini 4. Hal ini dilakukan dengan cara mencoba semua kemungkinan urutan yang memberikan *completion time* terendah yang berarti di sini *setup time* harus rendah.

Tabel 4.14 menunjukkan kemungkinan urutan di lini 1:

Tabel 4. 9 Kemungkinan Urutan *Job* pada Lini 1

Urutan			Total Setup	Pt	Completion Time
3	2	1	3.3	301	304.3
3	1	2	1.6		302.6
2	3	1	1.7		302.7
2	1	3	3.2		304.2
1	2	3	1.9		302.9
1	3	2	3.1		304.1

Maka dari itu urutan terbaik di lini 1 adalah: 3-1-2 karena memberikan *total setup time* terpendek yaitu 1.6 dengan *completion time* terpendek yaitu 302.6. Tabel 4.15 menunjukkan kemungkinan urutan di lini 4:

Tabel 4. 10 Kemungkinan Urutan *Job* pada Lini 4

Urutan		Total Setup	Pt	Completion Time
9	10	1.4	227.5	228.9
10	9	1.5		229

Maka dari itu urutan terbaik di lini 4 adalah: 9-10 karena memberikan *total setup time* terpendek yaitu 1.4 dengan *completion time* terpendek yaitu 228.9. Selanjutnya adalah menentukan urutan dari *job* yang memiliki opsi lini lebih dari 1 atau dalam Tabel 4.13 sebagai *sharejob*. Diawali dengan mengecek apakah jika semua *job* dikerjakan pada lini 2 menyebabkan terjadinya *overtime* atau tidak. *Completion time* jika semua dikerjakan di lini 2 dengan urutan tetap sesuai pada *sharejob* adalah 570.89 jam dengan *total setup time* 5.7. Maka dari itu masuk akan ada *job* yang dialokasikan ke lini 3 dengan menggunakan algoritma LPT untuk minimasi *makespan* sebagai berikut:

Step 1: Urutkan *job* dari *processing time* terpanjang dan Set total *completion time* lini 2 C2= 0 dan lini 3 C3=0

Job	4	5	6	8	7
Pt	308.82	92.65	67.5	51.22	45

Step 2: Masukkan *job* dengan *processing time* terpanjang ke lini 2 dan hitung C2

Lini	Job	Completion Time
Lini 2	4	308.82
Lini 3		0

Step 3: Masukkan *job* dengan *processing time* terpanjang berikutnya ke lini yang memiliki nilai *Ci* terendah dan *update* nilai *Ci*.

Lini	Job	Completion Time
Lini 2	4	308.82
Lini 3	5	92.65

Step 4: Ulangi step 2 dan 3 hingga semua *job* telah dimasukkan ke dalam lini

Lini	Job					Completion Time
Lini 2	4					308.82
Lini 3	5	6	8	7		256.37

Maka dari itu didapatkan urutan terbaik di lini 2 adalah 4 dan urutan terbaik di lini 3 adalah 5-6-8-7 dengan total *setup time* sebesar 4.5 jam. *Makespan* dari kedua lini ini adalah 308.82. Urutan ini dibuat tanpa mempertimbangkan *sequence dependent setup* terlebih dahulu. Kemudian akan masuk ke tahap pencarian struktur tetangga dengan cara melakukan pertukaran *job* antar lini terlebih dahulu kemudian dilanjutkan dengan pertukaran *job* dalam lini. Berikut adalah cara melakukan pencarian struktur tetangga:

Step 1: Tahap inisialisasi

Set T = *Makespan*

Maksimum iterasi = 5

Alpha (faktor penurunan suhu) = 0.2

Step 2: Tentukan secara random *job* yang akan ditukar antara lini 2 dan lini 3. Jika hasil penukaran lebih baik maka akan lanjut ke iterasi berikutnya. Jika hasil penukaran lebih buruk maka akan dibangkitkan bilangan random untuk dibandingkan dengan *probability Boltzman*.

Iterasi = 1

Makespan = 308.82

$T = 308.82$

Job tukar lini 2: 4

Job tukar lini 3: 5 – 8

Hasil akhir:

Lini	Job			Setup Time	Completion Time	Makespan_New
Lini 2	5	8		1.9	143.87	421.32
Lini 3	4	6	7	1.7	421.32	

Karena *makespan* hasil iterasi lebih buruk dari *initial solution* maka bangkitkan bilangan random $r = 0.86$. Hitung $\delta = \text{makespan_new} - \text{makespan}$ dan $PE = \exp(-\delta/T)$. Didapatkan $PE = 0.7257$. Jika bilangan random yang dihasilkan kurang dari PE maka hasil diterima, jika tidak maka hasil akan dikembalikan ke semula. Dalam hal ini karena $r > PE$ maka hasil dikembalikan ke sebelumnya.

Step 3: Set iterasi = iterasi + 1. Lakukan kembali step 2 hingga iterasi mencapai maksimum iterasi. Jika sudah mencapai maksimum iterasi maka lanjut ke step 4.

Step 4: Turunkan suhu dengan cara $T = T \cdot \alpha$. Maka T akan menjadi = 61.76. Kemudian set iterasi kembali ke 1 dan lakukan lagi dari step 2. Hal ini akan selesai ketika nilai T sudah mencapai sangat kecil mendekati nol.

Fase *constructive* belum menjamin memberikan hasil yang *global optimum*, maka dari itu diperlukannya fase *improvement* yang dapat memperbaiki hasil. Pada fase *constructive* mungkin menjamin memberikan nilai *makespan* yang pendek tetapi belum menjamin bahwa total *setup time* akan rendah. Hal ini dapat terlihat dari *initial solution* total *setup time*-nya adalah 4.5 jam sedangkan pada iterasi pertama total *setup time* adalah 3.1 jam. Tetapi *makespan* dari *initial solution* lebih baik dibandingkan dengan *makespan* iterasi 1. Maka dari itu akan dilakukan pencarian ke struktur tetangga yang dapat memberikan nilai baik untuk kedua parameter ini. Dapat dilihat pula bahwa penempatan *job* yang salah dalam lini

menyebabkan akumulasi *processing time* menjadi panjang, sehingga dapat dikatakan bahwa algoritma yang dibuat sekarang telah valid untuk mengalokasikan *job* dalam lini dengan tujuan minimasi *makespan* sekaligus *total setup time*.

Setelah melakukan penjadwalan untuk produk MTS maka selanjutnya adalah melakukan penjadwalan untuk produk MTO. Dalam proses perhitungan manual ini menggunakan 2 tambahan produk MTO yang disajikan dalam Tabel 4.16 di bawah ini:

Tabel 4. 11 Permintaan Produksi Produk MTO

<i>Job</i>	Ukuran Produk	Kg/jam	Permintaan Produksi (kg)	<i>Processing time</i> (dalam jam)	Opsi Lini	Tb
11	50 gram	114.29	22000	168.00	4	200
12	250 gram	485.71	35000	49.00	2,3,4	250

Produk dengan opsi lini hanya satu maka langsung dijadwalkan dengan mengecek apakah lini tersebut masih sedang melakukan proses produksi atau tidak. *Job* 11 hanya dapat diproduksi pada lini 4 dimana saat *job* 11 datang, lini 4 masih mengerjakan *job* 10. Maka dari itu urutan lini 4 menjadi 9-10-11. Sedangkan untuk *job* 12 yang memiliki opsi lebih dari 1 maka akan mengecek lini mana yang menyelesaikan proses produksi terlebih dahulu maka lini itulah yang terpilih.

4.4.2 Verifikasi dan Validasi Algoritma *Liquid*

Perbedaan utama algoritma *liquid* dan *powder* adalah cara pengalokasian *job* dalam lini. Perbedaan ini disebabkan karena lini produksi produk *liquid* memiliki kecepatan yang berbeda-beda sehingga pengalokasian produk yang salah dalam lini bisa mengakibatkan *processing time* dari *job* tersebut semakin panjang. Penerapan *Hungarian method* bertujuan untuk mengalokasikan *job* dalam lini produksi yang tepat sehingga dapat meminimasi *total completion time*-nya. Tabel 4.17 menunjukkan *processing time* produk dalam tiap lini *liquid* dengan kecepatan yang berbeda:

Tabel 4. 12 Permintaan Produksi Produk *Liquid*

Job	Ukuran Produk	Liter/ jam	Permintaan Produksi	Processing time (dalam jam)				Opsi Lini
				Lini 1	Lini 2	Lini 3	Lini4	
1	5000 ml	285.71	48000	168	9999	9999	9999	3
2	1000 ml	428.57	21000	49	9999	9999	9999	3
3	2500 ml	714.29	60000	84	80	88	9999	1,2,3
4	250 ml	485.71	150000	309	293	324	9999	1,2,3
5	250 ml	485.71	45000	93	88	97	9999	1,2,3
6	500 ml	800	54000	68	64	71	9999	1,2,3
7	500 ml	800	36000	45	43	47	9999	1,2,3
8	1000 ml	1171.43	60000	51	49	54	9999	1,2,3
9	60 ml	42.86	7500	9999	9999	9999	158	4
10	50 ml	114.29	6000	9999	9999	9999	31	4

Sama hal nya dengan produk *liquid*, untuk produk-produk yang hanya memiliki opsi lini sama dengan 1 maka yang dilakukan adalah mencoba semua kemungkinan urutan. Sedangkan untuk produk *liquid* yang memiliki opsi lebih dari satu maka diselesaikan dengan menggunakan *Hungarian Method* dengan bantuan *software* Lingo. Gambar 4.7 adalah *coding* dari *Hungarian Method*

```
sets:
    job/1..10/;;
    mesin/1..3/;;
    assign(job, mesin): processingtime, x;
endsets

data:
    processingtime=
    168    160    176
    49     47     51
    84     80     88
    309    293    324
    93     88     97
    68     64     71
    45     43     47
    51     49     54
    175    166    184
    53     50     55
;
@ole('C:\Users\G40\Desktop\excel penting\Monica_Scheduling Algorithm.xlsx','cij') = x;
enddata

min = @sum(assign(i,j): processingtime(i,j)*x(i,j));

@for(job(i): @sum(mesin(j): x(i,j)) =1);
@for(mesin(j): @sum(job(i): x(i,j)) >=1);
@sum(job(i): processingtime(i,1)*x(i,1)) <= 420;
@sum(job(i): processingtime(i,2)*x(i,2)) <= 420;
@sum(job(i): processingtime(i,3)*x(i,3)) <= 420;
@for(assign(i,j): @bin(x(i,j)));
```

Gambar 4. 5 *Hungarian Method* dengan *Software* Lingo

Kemudian di-*solve* dan tidak terjadi *error*, hal tersebut dibuktikan pada Gambar 4.8 yang menyebutkan bahwa problem ini termasuk dalam *Pure Integer Linear Programming* dengan status *Global Optimum* total *objective function* adalah

1084. Hasil *running* dari Lingo dimasukan ke dalam *excel* seperti tercantum pada Tabel 4.18.

Solver Status

Model Class: PIIP

State: Global Opt

Objective: 1084

Infeasibility: 0

Iterations: 0

Variables

Total: 30

Nonlinear: 0

Integers: 30

Constraints

Total: 17

Nonlinear: 0

Nonzeros

Total: 120

Nonlinear: 0

Extended Solver Status

Solver Type: B-and-B

Best Obj: 1084

Obj Bound: 1084

Steps: 0

Active: 0

Generator Memory Used (K)

27

Elapsed Runtime (hh:mm:ss)

00:00:00

Update Interval: 2

Interrupt Solver

Close

Gambar 4. 6 Status Solusi *Hungarian Method*

Tabel 4. 13 Solusi *Hungarian Method*

<i>Job</i>	Permintaan Produksi	Lini 1	Lini 2	Lini 3
1	48000	0	0	1
2	21000	0	0	1
3	60000	1	0	0
4	150000	1	0	0
5	45000	0	1	0
6	54000	0	1	0
7	36000	0	0	1
8	60000	0	1	0
9	7500	0	1	0
10	6000	0	1	0

Berdasarkan Tabel 4.18 dapat dilihat bahwa lini 2 mendapatkan alokasi *job* paling banyak dibandingkan dengan *job* lainnya. Hal ini karena lini 2 memiliki *processing time* tersingkat dibandingkan lini lainnya yang berarti kecepatannya lebih tinggi. Apabila salah satu produk dari lini 2 berpindah ke lini lainnya (ke lini yang memiliki kecepatan lebih rendah) akan menyebabkan *processing time*

semakin panjang (*objective function* semakin besar). Dengan demikian maka dapat dikatakan bahwa solusi dari *Hungarian method* yang digunakan telah sesuai dengan apa yang diharapkan.

BAB 5

UJI COBA DAN ANALISIS

Pada bab ini akan dibahas mengenai pengujian terhadap algoritma yang telah dibuat dengan menggunakan data yang sesungguhnya. Selain itu juga akan dilakukan analisis terhadap *output* dari algoritma yang telah dibuat.

5.1 Implementasi Algoritma

Dalam subbab ini akan dilakukan penerapan algoritma dengan menggunakan data dari PT Bayer Indonesia Divisi *Cropscience*. Data yang digunakan adalah data permintaan produksi, matriks *setup time*, *penalty overtime cost* dan kapasitas produksi tiap lini. Data-data tersebut telah dicantumkan pada bab 4 dan lampiran. *Overtime cost* diasumsikan Rp 35.000 per jam dan biaya penggunaan lini diasumsikan Rp 10.000.000.

5.1.1 Penjadwalan Produk *Powder* Per Bulan

Data permintaan produksi tidak dapat langsung diproses melainkan harus diubah dalam matriks produk-mesin yang berisi nilai 1 dan 0. Bernilai 1 apabila produk tersebut dapat diproduksi pada mesin yang bersangkutan dan bernilai 0 apabila sebaliknya. Berikut adalah matriks produk-mesin untuk tiap bulan:

Tabel 5. 1 Nomor Kode Produk *Powder*

Produk ke-	Kode Produk	Kode Negara		Produk ke-	Kode Produk	Kode Negara
1	T-50	ID		15	A -1 kg	ID
2	Al-250	ID		16	A -1 kg	MY
3	Al-250	MY		17	A -1 kg	PH
4	Al - 5 kg	ID		18	A -1 kg	PK
5	Al - 5 kg	PH		19	A- 10 kg	AU
6	Al - 10 kg	ID		20	A-2500	ID
7	C-100	ID		21	F-50	ID
8	IWP - 50	ID		22	F-50	PH
9	IWP - 50	BD		23	F-100	ID
10	IWP - 100	ID		24	F-250	ID
11	IWP - 100	BD		25	L-15	ID
12	IWP - 500	ID		26	L-100	ID
13	A-250	ID		27	L-100	BD
14	A-500	ID				

Tabel 5. 2 Matriks Produk-Mesin Bulan Maret dan April Produk *Powder*

Produk ke-	Maret				April			
	Lini 1	Lini 2	Lini 3	Lini 4	Lini 1	Lini 2	Lini 3	Lini 4
1	0	0	0	1	0	0	0	0
2	0	0	0	0	0	0	0	0
3	0	0	0	0	0	1	1	1
4	0	0	0	0	1	0	0	0
5	1	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0
7	0	0	0	1	0	0	0	1
8	0	0	0	0	0	0	0	0
9	0	0	0	1	0	0	0	0
10	0	0	0	1	0	0	0	0
11	0	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0	0
13	0	1	1	1	0	1	1	1
14	0	1	1	1	0	1	1	1
15	0	1	1	1	0	1	1	1
16	0	1	1	1	0	0	0	0
17	0	1	1	1	0	1	1	1
18	0	1	1	1	0	0	0	0
19	0	0	0	0	1	0	0	0
20	0	0	0	0	1	0	0	0
21	0	0	0	1	0	0	0	1
22	0	0	0	0	0	0	0	1
23	0	0	0	0	0	0	0	0
24	0	1	1	1	0	0	0	0
25	0	0	0	1	0	0	0	0
26	0	0	0	1	0	0	0	0
27	0	0	0	0	0	0	0	0

Tabel 5. 3 Matriks Produk-Mesin Bulan September dan Oktober Produk *Powder*

Produk ke-	September				October			
	Lini 1	Lini 2	Lini 3	Lini 4	Lini 3	Lini 4	Lini 3	Lini 4
1	0	0	0	0	0	0	0	1
2	0	0	0	0	0	0	0	0
3	0	0	0	0	0	1	1	1
4	0	0	0	0	0	0	0	0
5	0	0	0	0	1	0	0	0
6	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0
9	0	0	0	1	0	0	0	0
10	0	0	0	1	0	0	0	0
11	0	0	0	0	0	0	0	0
12	0	1	1	1	0	0	0	0
13	0	1	1	1	0	1	1	1
14	0	1	1	1	0	1	1	1
15	0	1	1	1	0	1	1	1
16	0	0	0	0	0	0	0	0
17	0	1	1	1	0	0	0	0
18	0	0	0	0	0	1	1	1

Tabel 5. 3 Matriks Produk-Mesin Bulan September dan Oktober Produk *Powder* (lanjutan)

Produk ke-	September				October			
	Lini 1	Lini 2	Lini 3	Lini 4	Lini 3	Lini 4	Lini 3	Lini 4
19	1	0	0	0	0	0	0	0
20	0	0	0	0	0	0	0	0
21	0	0	0	0	0	0	0	1
22	0	0	0	1	0	0	0	0
23	0	0	0	0	0	0	0	0
24	0	1	1	1	0	0	0	0
25	0	0	0	0	0	0	0	1
26	0	0	0	1	0	0	0	0
27	0	0	0	1	0	0	0	0

Kemudian dari matriks di atas akan dimasukan ke dalam Matlab sebagai salah satu input untuk diproses berdasarkan algoritma yang telah dikembangkan. Algoritma ini akan melakukan iterasi secara terus menerus hingga mencapai *stopping criteria* yang telah ditentukan. Setiap tahapan iterasi akan menyimpan hasil solusi sementara tetapi ketika ada solusi baru lebih buruk dari solusi sebelumnya tidak langsung dibuang melainkan akan ada peluang untuk menerima solusi buruk itu sehingga bisa mencapai hasil yang lebih baik. Implementasi algoritma untuk produk *powder* dilakukan pada dua keadaan yaitu ketika *peak season* dan *normal normal*.

5.1.1.1 Hasil Penjadwalan Bulan Maret

Berdasarkan data yang ada pada bab 4 mengenai *demand* produksi produk MTS, dilakukan tahap pertama algoritma yaitu menyusun produk dari M_j terkecil hingga M_j terbesar. Produk yang memiliki $M_j=1$ akan dikelompokkan dan dilakukan penjadwalan terlebih dahulu. Dimana jika produk masuk dalam kategori *dedicated lini 1* berarti produk tersebut hanya bisa diproduksi di lini 1 sedangkan *sharejob* berarti produk dapat diproduksi di lini 2 ataupun 3. Tabel 5.4 menunjukkan hasil dari algoritma *Least Flexible Job* yang telah dilakukan:

Tabel 5. 4 Hasil LFJ Produk *Powder* Bulan Maret

<i>Dedicated Lini 1</i>	5						
<i>Sharejob</i>	13	14	15	16	17	18	24
<i>Dedicated Lini 4</i>	1	7	9	10	21	25	26

Dari Tabel 5.4 di atas dapat dilihat bahwa terdapat 7 produk yang hanya bisa diproduksi di lini 4 dan 7 produk yang dapat diproduksi di lini 2 atau 3 dan 1 produk yang dapat diproduksi di lini 1. Tabel 5.5 dan Tabel 5.6 adalah hasil akhir dari algoritma yang dibuat:

Tabel 5. 5 Hasil Penjadwalan Produk *Powder* Bulan Maret

Lini Filling	Total setup (jam)	Completion time (jam)	Makespan (jam)	Indeks	Penalty Overtime Cost (Rp)	Biaya Lini (Rp)
1	0	108	366.4	4	0	30,000,000
2	3.95	350.95				
3	Tidak terpilih					
4	9.97	366.4				

Tabel 5. 6 Urutan Produk *Powder* dalam Lini bulan Maret

Lini Filling	Urutan Best						
1	5						
2	24	18	17	16	15	13	14
3	<i>Tidak terpilih</i>						
4	7	9	10	25	26	21	1

Dari tabel 5.6 di atas dapat dilihat bahwa lini 3 tidak terpilih untuk diproduksi karena seluruh produk dapat diproduksi dalam lini 2 tanpa melebihi kapasitas. Berikut adalah urutan kode produk berdasarkan tabel:

Lini 1 = A1-5kg (PH)

Lini 2 = F-250gr (ID) // A-1kg (PK) // A-1kg (PH) // A-1kg (MY) // A-1kg (ID) // A-250gr (ID) // A-500gr (ID)

Lini 4 = C-100gr (ID) // IWP-50gr (BD) // IWP-100gr (ID) // L-15gr (ID) // L-100gr (ID) // F-50gr (ID) // T-50gr (ID)

Data permintaan produksi untuk produk MTO didapatkan dari data historis perusahaan sedangkan data kedatangan produk MTO diasumsikan

telah mulai datang setelah hari ke 5 atau setelah 105 jam. Data tersebut ditampilkan pada tabel di bawah ini:

Tabel 5. 7 Data Produk MTO Produk *Powder* bulan Maret

Produk ke	Kode Produk	Kode Negara	Demand	PT (jam)	Kedatangan (jam)	Opsi Lini
1	T-50	ID	10975	96	105	4
3	Al-250	MY	42800	88	140	2,3
17	A -1 kg	PH	87550	75	150	2,3
19	A- 10 kg	AU	21460	50	200	1
6	Al - 10 kg	ID	14720	34	250	1

Berdasarkan tabel 5.7 dapat dilihat bahwa produk 1, 19 dan 6 adalah produk yang bersifat *dedicated* terhadap lininya sehingga ketika produk tersebut datang dapat langsung dicek apakah M1 dan M4 sedang melakukan proses produksi atau tidak. Jika tidak maka produk tersebut dapat langsung diproses. Berikut adalah hasil penjadwalan ketika produk MTO datang dimana pada Tabel 5.9 produk MTO ditandai dengan tulisan tebal dalam kotak abu-abu:

Tabel 5. 8 Hasil Penjadwalan Baru Produk *Powder* Bulan Maret

Lini Filling	Total setup (jam)	Completion time (jam)	Makespan (jam)	Indeks	Penalty Overtime Cost (Rp)	Biaya Lini (Rp)
1	4.38	288.38	467.99	4	1,715,000	40,000,000
2	3.95	350.95				
3	2.17	227.17				
4	13.99	467.99				

Tabel 5. 9 Urutan Baru Produk *Powder* dalam Lini bulan Maret

Lini Filling	Urutan Best							
1	5	19	6					
2	24	18	17	16	15	13	14	
3	3	17						
4	7	9	10	25	1b	26	21	1

5.1.1.2 Hasil Penjadwalan Bulan April

Berikut adalah hasil dari algoritma *Least Flexible Job* yang telah dilakukan berdasarkan Tabel 5.2:

Tabel 5. 10 Hasil LFJ Produk *Powder* Bulan April

Dedicated Lini 1	4	19	20		
Sharejob	3	13	14	15	17
Dedicated Lini 4	7	21	22		

Tabel 5. 11 Hasil Penjadwalan Produk *Powder* Bulan April

Lini Filling	Total setup (jam)	Completion time (jam)	Makespan (jam)	Indeks	Penalty Overtime Cost (Rp)	Biaya Lini (Rp)
1	2.88	224.88	322.28	2	0	40,000,000
2	1.28	322.28				
3	3.14	224.14				
4	3.3	293.3				

Tabel 5. 12 Urutan Produk *Powder* dalam Lini bulan April

Lini Filling	Urutan Best		
1	4	19	20
2	14	15	
3	3	13	17
4	22	21	7

Berikut adalah urutan kode produk berdasarkan Tabel 5.12:

Lini 1 = A1-5kg (ID) // A-10kg (AU) // // A-2500kg (ID)

Lini 2 = A-500gr (ID) // A-1kg (ID)

Lini 3 = A1-250gr (MY) // A-250gr (ID) // A-1kg (PH)

Lini 4 = F-50gr (ID) // F-50gr (PH) // C-100gr (ID)

Setelah penjadwalan awal dilakukan saat $t=0$ maka proses produksi dapat berjalan sesuai dengan jadwal yang telah dibuat sebelum produk MTO datang. Berikut adalah data produk MTO pada bulan April:

Tabel 5. 13 Data Produk MTO Produk *Powder* bulan April

Produk ke	Kode Produk	Kode Negara	Demand	PT (jam)	Kedatangan (jam)	Opsi Lini
13	A-250	ID	66238	136	105	2,3

Produk ke	Kode Produk	Kode Negara	Demand	PT (jam)	Kedatangan (jam)	Opsi Lini
14	A-500	ID	88320	271	140	2,3
15	A -1 kg	ID	180826	92	150	2,3

Berdasarkan Tabel 5.13 berikut adalah hasil penjadwalan ketika produk MTO datang dimana produk ke 13 MTO ditulis sebagai produk 13b dan produk MTO ditandai dengan tulisan dicetak tebal dalam kotak berwarna abu-abu:

Tabel 5. 14 Hasil Penjadwalan Baru Produk *Powder* Bulan April

Lini Filling	Total setup (jam)	Completion time (jam)	Makespan (jam)	Indeks	Penalty Overtime Cost (Rp)	Biaya Lini (Rp)
1	2.88	224.88	431	2	980,000	40,000,000
2	2.63	431				
3	3.14	429				
4	3.3	293.3				

Tabel 5. 15 Urutan Baru Produk *Powder* dalam Lini bulan April

Lini Filling	Urutan Best				
1	4	19	20		
2	14	15	14		
3	3	13	13b	17	15
4	22	21	7		

5.1.1.3 Hasil Penjadwalan Bulan September

Berikut adalah hasil dari algoritma *Least Flexible Job* yang telah dilakukan berdasarkan Tabel 5.3:

Tabel 5. 16 Hasil LFJ Produk *Powder* Bulan September

Dedicated Lini 1	19					
Sharejob	12	13	14	15	17	24
Dedicated Lini 4	9	10	22	26	27	

Tabel 5. 17 Hasil Penjadwalan Produk *Powder* Bulan September

Lini Filling	Total setup (jam)	Completion time (jam)	Makespan (jam)	Indeks	Penalty Overtime Cost (Rp)	Biaya Lini (Rp)
1	0	41	274.95	2	0	30,000,000

Lini Filling	Total setup (jam)	Completion time (jam)	Makespan (jam)	Indeks	Penalty Overtime Cost (Rp)	Biaya Lini (Rp)
2	4.95	274.95				
3	Tidak terpilih					
4	4.72	243.72				

Tabel 5. 18 Urutan Produk *Powder* dalam Lini bulan September

Lini Filling	Urutan Best					
1	19					
2	24	17	15	13	14	12
3	Tidak terpilih					
4	27	26	10	9	22	

Lini 1 = A-10kg (AU)

Lini 2 = F-250gr (ID) // A-1kg (PH) // A-1kg (ID) // A-250gr (ID) // A-500gr (ID) // IWP-500gr (ID)

Lini 4 = L-100gr (BD) // L-100gr (ID) // IWP-100gr (ID) // IWP-50gr (BD) // F-50gr (PH)

Setelah penjadwalan awal dilakukan saat $t=0$ maka proses produksi dapat berjalan sesuai dengan jadwal yang telah dibuat sebelum produk MTO datang. Berikut adalah data produk MTO pada bulan September:

Tabel 5. 19 Data Produk MTO Produk *Powder* bulan September

Produk ke	Kode Produk	Kode Negara	Demand	PT (jam)	Kedatangan (jam)	Opsi Lini
1	T-50	ID	6850	60	105	4
7	C-100	ID	2875	13	140	4
20	A-2500	ID	140000	196	150	1
24	F-250	ID	10200	21	200	2,3
5	Al - 5 kg	PH	8578	30	250	1

Berikut adalah hasil penjadwalan ketika produk MTO datang dimana produk MTO ditandai dengan tulisan bercetak tebal dalam kotak abu-abu:

Lini Filling	Total setup (jam)	Completion time (jam)	Makespan (jam)	Indeks	Penalty Overtime Cost (Rp)	Biaya Lini (Rp)
1	3.83	379.83	379.83	1	0	30,000,000
2	9.38	300.38				
3	Tidak terpilih					
4	10.76	323.76				

Lini Filling	Urutan Best						
1	19	20	5				
2	24	17	15	13	14	24	12
3	<i>Tidak terpilih</i>						
4	27	26	10	9	1	22	7

5.1.1.4 Hasil Penjadwalan Bulan Oktober

Berikut adalah hasil dari algoritma *Least Flexible Job* yang telah dilakukan berdasarkan Tabel 5.3:

Tabel 5. 20 Hasil LFJ Produk *Powder* Bulan Oktober

<i>Dedicated Lini 1</i>	5				
<i>Sharejob</i>	3	13	14	15	18
<i>Dedicated Lini 4</i>	1	21	25		

Tabel 5. 21 Hasil Penjadwalan Produk *Powder* Bulan Oktober

Lini Filling	Total setup (jam)	Completion time (jam)	Makespan (jam)	Indeks	Penalty Overtime Cost (Rp)	Biaya Lini (Rp)
1	0	79	372.33	2	0	30,000,000
2	4.33	372.33				
3	Tidak terpilih					
4	3.12	157.12				

Tabel 5. 22 Urutan Produk *Powder* dalam Lini bulan Oktober

Lini Filling	Urutan Best					
1	5					
2	3	13	14	18	15	
3	<i>Tidak terpilih</i>					
4	25	1	21			

Lini 1 = A1-5kg (PH)

Lini 2 = A1-250gr (MY) // A-250gr (ID) // A-500gr (ID) // A-1kg (PK) // A-1kg (ID)

Lini 4 = L-15gr (ID) // T-50gr (ID) // F-50gr (ID)

Gambar di bawah ini adalah *gantchart* penjadwalan bulan Oktober untuk produk *powder*. Pada gambar dapat dilihat bahwa lini 3 tidak terpilih

untuk melakukan proses produksi karena lini 2 mampu memproduksi keseluruhan produk MTS.

Setelah penjadwalan awal dilakukan saat $t=0$ maka proses produksi dapat berjalan sesuai dengan jadwal yang telah dibuat sebelum produk MTO datang. Berikut adalah data produk MTO pada bulan Oktober

Tabel 5. 23 Data Produk MTO Produk *Powder* bulan Oktober

Produk ke	Kode Produk	Kode Negara	Demand	PT (jam)	Kedatangan (jam)	Opsi Lini
12	IWP - 500	ID	5200	7	105	2,3
8	IWP - 50	ID	4320	38	140	4
13	A-250	ID	17090	35	150	2,3
16	A -1 kg	MY	150729	129	250	2,3
17	A -1 kg	PH	124500	56	200	2,3
6	Al - 10 kg	ID	50000	117	200	1

Berikut adalah hasil penjadwalan ketika produk MTO datang yang ditandai dengan nomor-nomor yang dicetak tebal dalam kotak berwarna abu-abu:

Tabel 5. 24 Hasil Penjadwalan Baru Produk *Powder* Bulan Oktober

Lini Filling	Total setup (jam)	Completion time (jam)	Makespan (jam)	Indeks	Penalty Overtime Cost (Rp)	Biaya Lini (Rp)
1	2.12	319.12	403	2	0	30,000,000
2	8.57	400.57				
3	0	403				
4	3.12	197.12				

Tabel 5. 25 Urutan Baru Produk *Powder* dalam Lini bulan Oktober

Lini Filling	Urutan Best					
1	5	6				
2	3	13	12	14	13b	16
3	17	18	15			
4	25	1	21	8		

Berikut adalah *gantchart* penjadwalan baru produk *powder* bulan Oktober yang telah mempertimbangkan MTO. Dalam gambar dapat dilihat bahwa setiap lini tidak ada yang melebihi kapasitas. Selain itu juga setelah kedatangan MTO lini 3 yang awalnya tidak terpilih menjadi terpilih. Hal ini karena *demand* yang cukup tinggi pada produk MTO menyebabkan lini 2 tidak lagi mampu memproduksi

seluruh produk. Di samping itu, lini 2 dan lini 3 sebagai *sharing* lini tetap dijaga keseimbangannya dan terlihat dalam gambar bahwa lini 2 dan lini 3 selesai dalam jangka waktu yang tidak berbeda secara signifikan.

5.1.2 Penjadwalan Produk *Liquid* Per Bulan

Dalam subbab ini akan ditampilkan hasil penjadwalan dari produk *liquid* bulan Maret, April, September dan Oktober. Dalam hasil penjadwalan akan ditampilkan nomor kode produk yang mana kode produk tersebut mengikuti daftar kode produk di bawah ini:

Tabel 5. 26 Nomor Kode Produk *Liquid*

Produk ke-	Kode Produk	Kode Negara	Produk ke-	Kode Produk	Kode Negara
1	Ag-500	ID	21	Bul-500	PH
2	Ag-1000	ID	22	Bul-1000	ID
3	Agr-1000	ID	23	Bul-1000	PH
4	Agr-5000	ID	24	Bul-200k	ID
5	Bc-100	ID	25	Cis -100	ID
6	Bc-500	ID	26	Cis -500	ID
7	Bf-250	ID	27	Cis -1000	ID
8	Bf-500	ID	28	CE-50	ID
9	Bf-5000	ID	29	CE-100	ID
10	Bi-500	ID	30	CSL-60	ID
11	Bi-500	PH	31	CSL-100	ID
12	Bi-1000	ID	32	CSL-250	ID
13	Bi-1000	PH	33	CSL-250	PK
14	BE-100	ID	34	CSL-500	ID
15	BE-250	ID	35	CSL-500	MY
16	BB-1000	ID	36	CSL-500	PH
17	BB-5000	ID	37	CSL-500	PK
18	Bul-100	ID	38	CSL-1000	ID
19	Bul-250	ID	39	CSL-1000	PK
20	Bul-500	ID	40	CSL-200k	ID

Tabel 5. 27 Nomor Kode Produk *Liquid* (Lanjutan)

Produk ke-	Kode Produk	Kode Negara	Produk ke-	Kode Produk	Kode Negara
41	D-50	ID	64	HSL-1000	ID
42	D-100	ID	65	HSL-1000	PH
43	D-250	ID	66	HSL-200k	ID
44	D-250	PH	67	HSL-200k	PH
45	D-500	ID	68	IMD-200k	ID
46	D-500	MY	69	ISC-100	ID
47	D-1000	ID	70	ISC-250	ID
48	D-1000	MY	71	CEW-1000	ID

Produk ke-	Kode Produk	Kode Negara	Produk ke-	Kode Produk	Kode Negara
49	D-1000	PH	72	LFS-100	ID
50	D-200k	ID	73	LFS-250	ID
51	ESL-100	ID	74	LFS-500	ID
52	ESL-1000	ID	75	MSC-100	ID
53	ESL-1000	TH	76	MSC-500	ID
54	ESL-5000	ID	77	PEV-20000	ID
55	ESL-5000	PH	78	PFT-20000	ID
56	ESL-20000	ID	79	Pre-60	ID
57	FSC-60	ID	80	Pre-250	ID
58	FSC-100	ID	81	PSL-100	ID
59	FSC-240	ID	82	PSL-250	ID
60	Fol-200k	ID	83	PSL-500	ID
61	GFS-50	ID	84	TSC-50	ID
62	GFS-100	ID	85	TSC-20000	ID
63	GFS-5000	ID			

Data permintaan produksi pada BAB 4 tidak dapat langsung diproses melainkan harus diubah dalam matriks produk-mesin yang berisi nilai 1 dan 0 seperti pada produk *powder*. Berikut adalah contoh matriks produk-mesin pada produk *liquid* yang mana keseluruhan matriks akan dicantumkan dalam lampiran:

Tabel 5. 28 Contoh Matriks Produk-Mesin Produk *Liquid*

Produk ke-	Maret				April			
	Lini 1	Lini 2	Lini 3	Lini 4	Lini 3	Lini 4	Lini 3	Lini 4
1	0	0	0	0	0	0	0	0
2	1	0	1	0	0	0	0	0
3	0	0	0	0	1	0	1	0
...
83	1	1	1	0	1	1	1	0
84	0	0	0	0	0	0	0	0
85	0	0	0	0	0	0	0	0

Kemudian dari matriks di atas akan dimasukkan ke dalam Matlab sebagai salah satu input untuk diproses berdasarkan algoritma yang telah dikembangkan. Implementasi algoritma untuk produk *liquid* dilakukan untuk 4 bulan.

5.1.2.1 Hasil Penjadwalan Bulan Maret

Berdasarkan data yang ada pada bab 4 mengenai *demand* produksi produk MTS, dilakukan tahap pertama algoritma yaitu menyusun produk dari M_j terkecil hingga M_j terbesar. Produk yang memiliki $M_j=1$ akan dikelompokkan dan dilakukan penjadwalan terlebih dahulu. Dimana jika

produk masuk dalam kategori *dedicated lini 4* berarti produk tersebut hanya bisa diproduksi di lini 4 begitu juga dengan *dedicated lini 3*. Sedangkan *sharejob13* berarti produk dapat diproduksi di lini 1 ataupun 3. Sedangkan *jobunig* berarti produk dapat diproduksi di lini 1,2 dan 3. Berikut adalah hasil dari algoritma *Least Flexible Job* yang telah dilakukan:

Tabel 5. 29 Hasil LFJ Produk *Liquid* Bulan Maret

<i>Dedlini 3</i>	9												
<i>Sharejob13</i>	2	13	23	49	65								
<i>Jobunig</i>	5	14	29	41	42	51	57	58	61	75	81	6	7
	8	11	15	20	21	43	44	45	83				
<i>Dedlini 4</i>	60	67	68										

Kemudian dilakukan penjadwalan langsung untuk produk-produk yang *dedicated* terhadap lininya. Setelah itu menentukan alokasi produk ke dalam lini untuk produk-produk dalam *sharejob*. Setelah itu maka akan dilakukan Hungarian Method untuk menentukan alokasi produk yang ada dalam *jobunig*. Permasalahan *assignment* dengan menggunakan Hungarian Method diselesaikan dengan menggunakan bantuan *software* Lingo. Hasil dari Lingo berada pada Tabel 5.30, yang mana tabel ini akan menjadi input untuk Matlab untuk memproses pencarian solusi tetangga yang telah mempertimbangkan *setup time*. Berikut adalah hasil dari Hungarian Method pada bulan Maret:

Tabel 5. 30 Hasil Hungarian Method bulan Maret

	Lini 1	Lini 2	Lini 3
5	1	0	0
6	0	0	1
7	0	0	1
8	0	0	1
11	0	0	1
14	1	0	0
15	0	0	1
20	0	0	1
21	0	0	1
29	1	0	0
41	1	0	0

	Lini 1	Lini 2	Lini 3
42	1	0	0
43	0	0	1
44	0	0	1
45	0	0	1
51	1	0	0
57	1	0	0
58	1	0	0
61	1	0	0
75	1	0	0
81	1	0	0
83	0	0	1

Berikut ada hasil akhir penjadwalan yang dilakukan:

Tabel 5. 31 Hasil Penjadwalan Produk *Liquid* Bulan Maret

Lini Filling	Total setup (jam)	Completion time (jam)	Makespan (jam)	Indeks	Penalty Overtime Cost	Biaya Lini (Rp)
1	17.75	199.75	226.61	3	0	30,000,000
2	Tidak terpilih					
3	16.61	226.61				
4	3.44	38.44				

Tabel 5. 32 Urutan Produk *Liquid* dalam Lini bulan Maret

Lini Filling	Urutan Best													
1	49	13	5	14	29	41	42	51	57	58	61	75	81	
2	<i>Tidak terpilih</i>													

Tabel 5.32 Urutan Produk *Liquid* dalam Lini bulan Maret (lanjutan)

Lini Filling	Urutan Best															
3	9	65	23	2	6	7	15	11	8	20	21	43	44	45	83	
4	60	67	68													

Gambar di bawah ini adalah *gantchart* produk *liquid* pada bulan Maret. Dari gambar di bawah dapat dilihat bahwa lini 2 tidak digunakan. Hal ini disebabkan karena penggunaan lini 3 dari perhitungan masih cukup untuk memproduksi seluruh produk MTS. Pemilihan lini dalam lini *sharing* diharapkan mampu meminimasi biaya penggunaan lini.

Setelah penjadwalan awal dilakukan saat $t=0$ maka proses produksi dapat berjalan sesuai dengan jadwal yang telah dibuat sebelum produk MTO datang. Berikut adalah data produk MTO pada bulan Maret:

Tabel 5. 33 Data Produk MTO Produk *Liquid* bulan Maret

Produk ke	Kode Produk	Kode Negara	Demand	Pt (jam)		Kedatangan (jam)	Ops Lini
				Lini 1/3	Lini 2		
3	Agr-1000	ID	50196	22	999	105	1,3
4	Agr-5000	ID	73820	50	999	140	3
8	Bf-500	ID	41070	28	39	150	1,2,3
9	Bf-5000	ID	10200	7	999	250	3

Berikut adalah hasil penjadwalan ketika produk MTO datang dimana produk MTO ditampilkan dalam tulisan yang dicetak tebal dalam kotak berwarna abu-abu:

Tabel 5. 34 Hasil Penjadwalan Baru Produk *Liquid* Bulan Maret

Lini Filling	Total setup (jam)	Completion time (jam)	Makespan (jam)	Indeks	Penalty Overtime Cost (Rp)	Biaya Lini (Rp)
1	19.35	229.35	308.99	3	0	30,000,000
2	Tidak terpilih					
3	19.99	308.99				
4	3.44	38.44				

Tabel 5. 35 Urutan Baru Produk *Liquid* dalam Lini bulan Maret

Lini Filling	Urutan Best															
1	49	13	5	14	29	41	42	51	57	58	61	8	75	81		
2	Tidak terpilih															
3	9	65	23	2	6	7	15	11	3	8	4	20	21	43		
	44	45	9	83												
4	60	67	68													

5.1.2.2 Hasil Penjadwalan Bulan April

Berikut adalah hasil dari algoritma *Least Flexible Job* yang telah dilakukan berdasarkan matriks produk-mesin lini produksi produk *liquid* (tabel dicantumkan dalam lampiran):

Tabel 5. 36 Hasil LFJ Produk *Liquid* Bulan April

Dedlini 3	9	55												
Sharejob13	3	23	39	48	49									
Jobunig	5	14	28	29	30	41	42	57	58	72	81	6		
	7	8	15	19	20	33	35	37	43	45	70	73	83	
Dedlini 4	Tidak terpilih													

Setelah mengetahui produk mana yang dapat dikerjakan dalam lini maka kemudian untuk lini yang bersifat *sharing* diselesaikan dengan menggunakan *Hungarian Method*. Berikut adalah hasil dari *running software Lingo*:

Tabel 5. 37 Hasil *Hungarian Method* bulan April

	Lini 1	Lini 2	Lini 3		Lini 1	Lini 2	Lini 3
5	1	0	0	37	0	1	0
6	0	0	1	41	1	0	0
7	0	0	1	42	1	0	0
8	0	0	1	43	0	0	1

	Lini 1	Lini 2	Lini 3
14	1	0	0
15	0	1	0
19	0	1	0
20	0	0	1
28	0	1	0
29	0	1	0
30	1	0	0
33	0	1	0
35	0	1	0

	Lini 1	Lini 2	Lini 3
45	1	0	0
57	1	0	0
58	0	1	0
70	1	0	0
72	0	1	0
73	1	0	0
81	1	0	0
83	1	0	0

Berikut ada hasil akhir penjadwalan yang dilakukan untuk semua lini:

Tabel 5. 38 Hasil Penjadwalan Produk *Liquid* Bulan April

Lini Filling	Total setup (jam)	Completion time (jam)	Makespan (jam)	Indeks	Penalty Overtime Cost	Biaya Lini
1	14.36	259.36	321.98	2	0	30,000,000
2	13.98	321.98				
3	12	187				
4	Tidak terpilih					

Tabel 5. 39 Urutan Produk *Liquid* dalam Lini bulan April

Lini Filling	Urutan Best													
1	39	81	73	70	23	49	33	35	37	42	45	57	83	
2	15	30	28	29	5	19	14	41	58	72				
3	55	9	48	3	6	7	8	20	43					
4	Tidak terpilih													

Gambar di bawah ini merupakan *ganttchart* penjadwalan bulan produk *liquid* bulan April. Dapat dilihat dari gambar bahwa penjadwalan pada bulan April tidak ada yang melebihi kapasitas produksi.

Setelah penjadwalan awal dilakukan saat $t=0$ maka proses produksi dapat berjalan sesuai dengan jadwal yang telah dibuat sebelum produk MTO datang. Berikut adalah data produk MTO pada bulan April:

Tabel 5. 40 Data Produk MTO Produk *Liquid* bulan April

Produk ke	Kode Produk	Kode Negara	Demand	Pt (jam)		Kedatangan (jam)	Ops Lini
				Lini 1/3	Lini 2		
8	Bf-500	ID	33020	23	31	105	1,2,3
6	Bc-500	ID	80150	54	75	140	1,2,3
29	CE-100	ID	47170	90	69	150	1,2

Berikut adalah hasil penjadwalan ketika produk MTO datang yang ditandai dengan nomor bercetak tebal:

Tabel 5. 41 Hasil Penjadwalan Baru Produk *Liquid* Bulan April

Lini Filling	Total setup (jam)	Completion time (jam)	Makespan (jam)	Indeks	Penalty Overtime Cost (Rp)	Biaya Lini (Rp)
1	15.26	351.09	351.9	1	0	30,000,000
2	13.98	321.98				
3	15.07	295.07				
4	Tidak terpilih					

Tabel 5. 42 Urutan Baru Produk *Liquid* dalam Lini bulan April

Lini Filling	Urutan Best													
1	39	81	73	70	23	49	33	35	37	42	29	45	83	
2	15	30	28	29	5	19	14	41	58	72				
3	55	9	48	3	6	7	8	8b	20	6	43	57		
4	Tidak terpilih													

5.1.2.3 Hasil Penjadwalan Bulan September

Berikut adalah hasil dari algoritma *Least Flexible Job* yang telah dilakukan berdasarkan Tabel Produk-Mesin yang telah dibuat (dicantumkan dalam lampiran):

Tabel 5. 43 Hasil LFJ Produk *Liquid* Bulan September

<i>Dedlini 3</i>	55														
<i>Sharejob13</i>	27	48	49	53											
<i>Jobunig</i>	18	29	30	41	42	61	69	81	19	20	43	44	45	46	65
<i>Dedlini 4</i>	65														

Kemudian dari hasil *sharejob13* dan *jobunig* tersebut akan menjadi input untuk melakukan *Hungarian Method* dengan menggunakan Lingo. Berikut adalah hasil akhir penjadwalan yang dilakukan:

Tabel 5. 44 Hasil Penjadwalan Produk *Liquid* Bulan September

Lini Filling	Total setup (jam)	Completion time (jam)	Makespan (jam)	Indeks	Penalty Overtime Cost (Rp)	Biaya Lini (Rp)
1	8.69	252.69	252.69	1	0	

Lini Filling	Total setup (jam)	Completion time (jam)	Makespan (jam)	Indeks	Penalty Overtime Cost (Rp)	Biaya Lini (Rp)
2	Tidak terpilih					30,000,000
3	7.95	105.95				
4	0	7				

Tabel 5. 45 Urutan Produk *Liquid* dalam Lini bulan September

Lini Filling	Urutan Best									
1	29	61	69	30	48	81	42	41	18	81
2	Tidak terpilih									
3	19	49	43	44	27	20	55	53	45	46
4	65									

Setelah penjadwalan awal dilakukan saat $t=0$ maka proses produksi dapat berjalan sesuai dengan jadwal yang telah dibuat sebelum produk MTO datang. Berikut adalah data produk MTO pada bulan September:

Tabel 5. 46 Data Produk MTO Produk *Liquid* bulan September

Produk ke	Kode Produk	Kode Negara	Demand	Pt (jam)		Kedatangan (jam)	Ops Lini
				Lini 1/3	Lini 2		
3	Agr-1000	ID	18228	8	999	105	1,3

Tabel 5.47 Data Produk MTO Produk *Liquid* bulan September (lanjutan)

Produk ke	Kode Produk	Kode Negara	Demand	Pt (jam)		Kedatangan (jam)	Ops Lini
				Lini 1/3	Lini 2		
4	Agr-5000	ID	31300	22	999	140	3
6	Bc-500	ID	38025	26	36	150	1,2,3

Berikut adalah hasil penjadwalan ketika produk MTO datang dimana produk MTO ditandai dengan tulisan bercetak tebal dalam kotak berwarna abu-abu:

Tabel 5. 47 Hasil Penjadwalan Baru Produk *Liquid* Bulan September

Lini Filling	Total setup (jam)	Completion time (jam)	Makespan (jam)	Indeks	Penalty Overtime Cost (Rp)	Biaya Lini (Rp)
1	8.69	252.69	252.69	1	0	30,000,000
2	Tidak terpilih					

Lini Filling	Total setup (jam)	Completion time (jam)	Makespan (jam)	Indeks	Penalty Overtime Cost (Rp)	Biaya Lini (Rp)
3	14.7	199.4				
4	0	7				

Tabel 5. 48 Urutan Baru Produk *Liquid* dalam Lini bulan September

Lini Filling	Urutan Best										
1	29	61	69	30	48	81	42	41	18	81	
2	Tidak terpilih										
3	19	49	43	44	27	20	55	53	45	46	3 4 6
4	65										

5.1.2.4 Hasil Penjadwalan Bulan Oktober

Berikut adalah hasil dari algoritma *Least Flexible Job* yang telah dilakukan berdasarkan tabel produk-mesin yang telah dibuat (dicantumkan dalam lampiran):

Tabel 5. 49 Hasil LFJ Produk *Liquid* Bulan Oktober

Dedlini 3	4	9	55								
Sharejob13	3	49	53								
Jobunig	5	28	29	41	42	51	61	75	6	8	
	15	20	26	33	35	36	37	44	45	70	76
Dedlini 4	Tidak terpilih										

Setelah mengetahui hasil tersebut maka akan lanjut ke tahap selanjutnya yaitu *Hungarian Method* untuk lini yang bersifat *sharing*. Proses perhitungan dilakukan dengan bantuan *software* Lingo. Berikut adalah hasilnya:

Tabel 5. 50 Hasil Hungarian Method bulan Oktober

	Lini 1	Lini 2	Lini 3
5	1	0	0
6	0	0	1
8	0	0	1
15	0	0	1
20	0	0	1
26	0	0	1
28	1	0	0
29	1	0	0
33	0	0	1
35	0	0	1

	Lini 1	Lini 2	Lini 3
36	0	0	1
37	0	0	1
41	1	0	0
42	1	0	0
44	0	0	1
45	0	0	1
51	1	0	0
61	1	0	0
70	0	0	1
75	1	0	0
76	0	0	1

Berikut adalah hasil akhir penjadwalan yang dilakukan:

Tabel 5. 51 Hasil Penjadwalan Produk *Liquid* Bulan Oktober

Lini Filling	Total setup (jam)	Completion time (jam)	Makespan (jam)	Indeks	Penalty Overtime Cost (Rp)	Biaya Lini (Rp)
1	9.47	135.47	149.64	3	0	20,000,000
2	Tidak terpilih					
3	16.64	149.64				
4	Tidak terpilih					

Tabel 5. 52 Urutan Produk *Liquid* dalam Lini bulan Oktober

Lini Filling	Urutan Best																
1	28	29	5	49	42	41	75	61	51	53							
2																	
3	3	4	20	76	15	8	55	37	36	35	33	26	9	45	44	70	6
4																	

Setelah penjadwalan awal dilakukan saat $t=0$ maka proses produksi dapat berjalan sesuai dengan jadwal yang telah dibuat sebelum produk MTO datang. Berikut adalah data produk MTO pada bulan Oktober:

Tabel 5. 53 Data Produk MTO Produk *Liquid* bulan Oktober

Produk ke	Kode Produk	Kode Negara	Demand	Pt (jam)		Kedatangan (jam)	Ops Lini
				Lini 1/3	Lini 2		
6	Bc-500	ID	13520	10	13	105	1,2,3
8	Bf-500	ID	176650	119	165	140	1,2,3
82	PSL-250	ID	18620	22	28	150	1,2,3

Berikut adalah hasil penjadwalan dengan produk MTO dimana produk MTO ditandai dengan tulisan bercetak tebal dalam kotak berwarna abu-abu:

Tabel 5. 54 Hasil Penjadwalan Baru Produk *Liquid* Bulan Oktober

Lini Filling	Total setup (jam)	Completion time (jam)	Makespan (jam)	Indeks	Penalty Overtime Cost (Rp)	Biaya Lini (Rp)
1	11.92	266.92	266.92	1	0	20,000,000
2	Tidak terpilih					
3	18.51	208.87				
4	Tidak terpilih					

Tabel 5. 55 Urutan Baru Produk *Liquid* dalam Lini bulan Oktober

Lini Filling	Urutan Best											
1	28	29	5	49	42	41	75	6	61	51	53	8
2	<i>Tidak terpilih</i>											
3	3	4	20	76	15	8	55	37	36	35		
	33	26	9	45	44	70	6	82				
4	<i>Tidak terpilih</i>											

5.1.3 Rekap Hasil Penjadwalan *Powder* dan *Liquid*

Dalam subbab ini akan ditampilkan rekapitulasi hasil penjadwalan produk *powder* dan *liquid*. Selain itu juga ditampilkan data *duration loss* berdasarkan *database* OPORT yang dimiliki oleh PT Bayer Indonesia Divisi *Cropscience*. Dari subbab ini akan dilihat seberapa baik penjadwalan yang telah dilakukan dengan membandingkan total *duration loss* dengan total *setup time* dan juga membandingkan *penalty overtime* hasil perhitungan dengan algoritma dengan *penalty overtime* sistem nyatanya. Secara keseluruhan hasil penjadwalan pada produk *powder* memberikan perbaikan sebesar 9.47% meskipun penjadwalan pada bulan Oktober lebih buruk dibandingkan dengan penjadwalan sistem *realnya*. Dalam subbab analisis akan dijelaskan mengapa hal ini dapat terjadi. Sedangkan hasil penjadwalan pada produk *powder* memberikan perbaikan sebesar 17.56% meskipun penjadwalan pada bulan September lebih buruk dibandingkan dengan penjadwalan sistem *realnya*.

5.2 Analisis Algoritma

Pada subbab ini akan dijelaskan mengenai analisa terhadap algoritma yang dibuat yaitu algoritma *powder* dan algoritma *liquid*. Analisa dilakukan terhadap *output* yang telah dihasilkan. Selain itu juga dilakukan analisa sensitivitas dengan cara merubah parameter-parameter tertentu.

5.2.1 Analisa Solusi Penjadwalan Produk *Powder* dan Produk *Liquid*

Berdasarkan penjadwalan yang telah dilakukan pada bulan Maret, April, September dan Oktober pada produk *liquid* dan *powder* didapatkan bahwa kemungkinan terjadi *overtime* besar pada keadaan *peak season* (Bulan Maret dan April). Hal ini dapat dilihat pada produk *powder* bulan Maret dan April memberikan *penalty overtime* yang disebabkan karena kuantitas dari *demand* yang tinggi. Sedangkan pada bulan September dan Oktober terdapat lini yang dari awal hingga akhir tidak digunakan karena permintaan produksi dapat dipenuhi hanya dengan tiga lini saja. Semakin banyak *demand* maka semua lini akan terpilih untuk dapat memproduksi *demand* dengan harapan *completion time* tiap lini tidak melebihi kapasitas. Dalam upaya membuat utilitas tiap mesin tidak mengalami ketimpangan, maka fungsi tujuan minimasi *makespan* dipilih agar dapat memberikan beban produksi yang tidak timpang antara lini satu dengan lini yang lainnya. Fungsi minimasi *makespan* juga berupaya agar *completion time* tiap lini tidak melebihi batas maksimal produksinya.

Selain itu dari penjadwalan yang dilakukan juga dapat dilihat bahwa urutan produk yang dihasilkan cenderung mengikuti jenis produknya. Contohnya adalah pada bulan Maret produk *powder* di lini 2 terlihat bahwa produk tipe A terkumpul menjadi 1 dalam 1 lini. Hal ini dapat terjadi karena mengelompokkan produk tipe yang sama dalam 1 lini dapat mengurangi terjadinya *setup* perubahan produk yang cenderung memakan waktu lebih lama apabila dibandingkan dengan pergantian ukuran produk. Pada bulan Maret produk *powder* di lini 2 diawali dengan produk F-250 gram kemudian A-1kg (PK) diikuti dengan produk yang sama untuk negara yang berbeda yaitu A-1kg(PH), A-1kg(MY) kemudian baru diikuti oleh A-1kg(ID). Ketika produk yang sama diurutkan seperti ini maka total setupnya adalah 0. Matriks *setup time* perlu dirancang sesesuai mungkin dengan keadaan aslinya

sehingga penentuan urutan produk dalam lini benar-benar dapat mengurangi total waktu *setup* nya. Penjadwalan yang dihasilkan pada kedua produk ini juga telah terjamin sesuai dengan aturan ACL (*Acceptance Contamination Limit*). Dari semua penjadwalan dapat dilihat bahwa *total setup time* menghasilkan angka yang wajar yang berarti angka 9999 tidak pernah terpilih. Untuk mencegah aturan ACL dilanggar maka dalam matriks *setup time* diberikan angka 9999 sehingga tidak akan pernah memilih produk yang dilarang dalam aturan ACL.

Pada Tabel 5.57 dapat dilihat bahwa penjadwalan pada bulan Oktober memberikan total setup yang lebih buruk yaitu lebih tinggi 6.64%. Pada Tabel 5.58 penjadwalan yang dilakukan pada bulan September juga memberikan total setup yang lebih buruk yaitu lebih tinggi 2.65%. Total *setup time* yang lebih buruk ini memungkinkan terjadi karena data *setup* yang digunakan adalah rata-rata dari proses *setup* yang dari tahun 2014 hingga 2018. Rata-rata bisa memberikan total waktu *setup* yang lebih tinggi dibandingkan dengan total waktu *setup* yang terjadi pada bulan September dan Oktober 2018.

5.2.2 Analisa Perbandingan *Duration Loss* dan *Penalty Overtime*

Salah satu tujuan dari penelitian ini adalah mampu membuat algoritma penjadwalan yang baik sehingga mampu mengurangi *total duration loss* yang mana merupakan penyebab rendahnya nilai availabilitas. Setelah melakukan proses *running* kemudian akan dibandingkan dengan data *duration loss* yang dimiliki oleh PT Bayer Indonesia Divisi *Cropscience*. Data *duration loss* yang diambil adalah kategori *format change*, *cleaning for product change*, *minor adjustment*, *waiting for ACL analysis* dan *emptying machine*. Kelima faktor dipilih karena merupakan faktor-faktor yang memiliki keterkaitan dengan proses *setup* yang terjadi. Dari hasil *running* Tabel 5.57 dapat dilihat bahwa algoritma penjadwalan yang dibuat untuk produk *powder* mampu mengurangi total *setup time* sebanyak 9.47% sedangkan pada Tabel 5.58 algoritma penjadwalan untuk produk *liquid* mampu mengurangi total *setup time* sebanyak 17.56%. Perbaikan terhadap total *setup time* ini dapat terjadi karena pada algoritma yang dibuat telah menerapkan pertimbangan *sequence dependent setup time* dan juga terdapat fase *improvement* dari *initial solution* sehingga menyebabkan hasil penjadwalan memberikan total *setup time* yang lebih

rendah. Hasil penjadwalan untuk produk *liquid* memberikan hasil yang lebih baik dari produk *powder* karena pada produk *liquid* variasi ukuran sangat banyak yaitu lebih dari 50. Tentunya hal ini akan sangat sulit bagi pihak PT Bayer Indonesia untuk menentukan urutan produk yang hendak diproduksi sehingga menyebabkan total *duration loss* sangat tinggi apabila dibandingkan dengan produk *powder*. Selain dari segi *setup time*, *penalty overtime* juga dapat dikurangi sebanyak 35.39%. Pada produk *powder* memiliki *overtime* yang lebih banyak karena sifat permintaan produksi pada produk *powder* adalah variasi rendah tetapi kuantitas tinggi. Kuantitas tinggi dapat menyebabkan *processing time* panjang karena proses produksi untuk satu produk tidak dapat dipecah. *Overtime* yang tinggi juga terjadi pada bulan Maret dan April yang mana merupakan *peak season*. Sedangkan pada bulan September dan Oktober, algoritma penjadwalan berhasil menjadwalkan sebaik mungkin sehingga tidak terjadi *overtime*.

5.2.3 Analisa Sensitivitas

Dalam subbab ini akan dijelaskan mengenai analisa sensitivitas yang dilakukan dengan cara mengubah paramater-parameter yang dapat mempengaruhi solusi dari algoritma. Analisa sensitivitas dibagi menjadi 3 yaitu perubahan penjadwalan akibat mesin rusak, perubahan permintaan produksi dan perubahan pada faktor penurun suhu dan jumlah iterasi pada fase *improvement*. Analisis sensitivitas ini bertujuan untuk melakukan pengecekan ulang terhadap model yang telah dibuat. Ketika sedang melakukan analisa sensitivitas pada suatu parameter, parameter lainnya tidak akan dirubah atau tetap.

5.2.3.1 Analisa Perubahan Penjadwalan Produksi akibat Mesin Rusak

Penjadwalan pada produk *liquid* dan *powder* sama-sama memiliki 4 lini *filling* yang bisa digunakan untuk memproduksi berbagai variasi produk. Dalam analisis sensitivitas ini skenario yang diterapkan adalah salah satu lini yang bersifat *sharing* rusak sehingga mengharuskan algoritma mengalihkan alokasi produk ke lini *filling* lain. Contohnya pada produk *powder*, lini yang bersifat *sharing* adalah lini 2 dan 3 sedangkan pada produk *liquid*, lini yang bersifat *sharing* adalah lini 1,2 dan 3 dengan lini 2 memiliki

kecepatan yang berbeda. Dalam hal ini lini *liquid filling* 1 akan diasumsikan rusak sehingga mempengaruhi alokasi produk dalam lini. Tahapan yang perlu dilakukan untuk mengatasi permasalahan ini adalah:

1. Pada tahapan *Hungarian method*, lini 1 diberi nilai 999 yang berarti model akan mencegah untuk mengalokasikan produk untuk diproduksi di lini 1.
2. Penentuan alokasi produk pada *Hungarian Method* tidak mempertimbangkan waktu *setup* melainkan hanya total *processing time* maka dari itu dilakukan pencarian struktur tetangga yang dapat meminimasi total waktu *setup*.
3. Pencarian struktur tetangga dilakukan dengan menukar alokasi produk lini *filling* 2 dengan lini *filling* 3. Selain itu juga dilakukan pencarian struktur tetangga dengan mengubah urutan produk dalam lini sendiri.

Berikut ini adalah hasil dari penjadwalan produksi setelah lini *filling* 1 diasumsikan tidak dapat digunakan karena rusak pada bulan April tanpa adanya kedatangan dari produk MTO:

Tabel 5. 56 Hasil Penjadwalan Skenario Lini 1 Rusak

Lini Filling	Total setup	Completion time	Makespan	Indeks	POC (Rp)	Biaya Lini (Rp)
1	Rusak		522.47	2	3,675,000	30,000,000
2	11.47	522.47				
3	19.90	313.9				
4	Tidak digunakan					
	31.37					

Berdasarkan hasil di atas dapat dilihat bahwa lini 2 akan terpilih meskipun kecepatan proses lini *filling* 2 lebih lama dibandingkan dengan lini *filling* 3. Pada keadaan awal dapat dilihat bahwa lini *filling* 2 tidak pernah terpilih karena *processing time* di lini *filling* 2 panjang. Selama kapasitas produksi lini *filling* 1 dan 3 masih mencukupi, Hungarian Method akan mengalokasikan produk ke kedua lini tersebut agar *makespan* tidak terlalu terpanjang. Namun ketika lini *filling* 1 rusak, *Hungarian Method* akan mengalokasikan produk ke lini *filling* 2 dan ke lini *filling* 3. Hal ini

menyebabkan *makespan* yang awalnya 321.98 dengan total *setup* 40.34 menjadi 522.47 dengan total *setup* 31.37. *Makespan* yang panjang ini menyebabkan penjadwalan produksi terkena *penalty overtime*. *Makespan* yang panjang ini dikarenakan penjadwalan produksi hanya bisa dilakukan pada 2 lini sehingga penggunaan lini 2 dan 3 dimaksimalkan. Namun permasalahan yang dapat terjadi berikutnya adalah ketika terdapat produk MTO datang dan lini *filling* 1 masih dalam keadaan rusak. Hal ini dapat mengakibatkan *makespan* akan semakin panjang yang berarti *penalty overtime* akan semakin besar.

Jika kerusakan yang terjadi adalah kerusakan pada lini *filling* yang bersifat *dedicated* terhadap produknya (seperti lini *filling* 4 pada lini produksi *powder* dan *liquid*) maka proses produksi untuk produk-produk tersebut tidak dapat dilakukan. Hal ini karena tidak ada lini lain yang dapat memproduksi produk tersebut sehingga langkah yang dapat dilakukan adalah menunggu hingga lini *filling dedicated* selesai diperbaiki.

5.2.3.2 Pengaruh Perubahan Permintaan Produksi terhadap Solusi Algoritma

Permintaan produksi mempengaruhi panjangnya akumulasi *processing time*, maka dari itu pada analisis sensitivitas ini akan dilihat bagaimana pengaruh perubahan permintaan produksi terhadap solusi dari algoritma. Permintaan produksi diubah jumlahnya menjadi setengah kali, satu setengah kali dan dua kali lipat dari jumlah permintaan awal. Dari perubahan jumlah permintaan produksi ini akan dilihat bagaimana pengalokasian produk dalam lini *filling sharing* sehingga dapat meminimasi *makespan* dengan harapan hal ini dapat meminimasi terjadinya *penalty overtime*. Apabila pengalokasian tidak tepat maka dapat menyebabkan *makespan* semakin panjang. *Overtime* dapat mudah terjadi pada lini yang bersifat *dedicated* apabila *demand* dinaikkan. Hal ini karena tidak ada lini lain yang mampu memproduksi produk tersebut sehingga hanya penentuan urutan dalam lini saja yang mempengaruhi panjangnya

completion time. Penentuan urutan dalam lini *dedicated* mempengaruhi total *setup time* sehingga mempengaruhi *completion time* lini *dedicated*.

Analisa sensitivitas dilakukan untuk penjadwalan bulan Maret produk *powder*. Berikut adalah hasil dari perubahan permintaan produksi dimana N1 merupakan jumlah produk yang dialokasikan pada lini 2 dan N2 merupakan jumlah produk yang dialokasikan pada lini 3:

Tabel 5. 57 Pengaruh Permintaan Produksi terhadap Solusi

Permintaan Produksi	N1	N2	Total Setup	Comp 2	Comp 3	Makespan	Penalty Overtime (Rp)
0.5*Keadaan Awal	7	0	12.35	176.95	0	188.97	-
Keadaan Awal	7	0	12.35	350.95	0	366.4	-
1.5*Keadaan Awal	4	3	13.82	261.47	263.38	546.97	4,655,000
2*Keadaan Awal	4	3	13.82	348	346	715	10,535,000
3*Keadaan Awal	4	3	13.82	520	522	1072	23,030,000

Dari hasil di atas dapat dilihat bahwa apabila lini *filling* 2 masih mampu mengerjakan produksi tanpa melebihi kapasitas maka semua produk akan dialokasikan pada lini *filling* tersebut. Penggunaan hanya 1 lini ini dapat meminimasi biaya penggunaan lini. Sedangkan ketika lini *filling* 2 tidak mampu mengerjakan proses produksi tanpa melebihi kapasitas maka algoritma akan melakukan proses alokasi produk dalam lini yang dapat meminimasi makespan. Dapat dilihat bahwa *completion* 2 dan *completion* 3 tidak berbeda secara signifikan yang menunjukkan bahwa algoritma telah melakukan alokasi produk dalam lini dengantepat sehingga dapat meminimasi *makespan*. Selain itu dapat dilihat pula bahwa *total setup* bernilai sama yang berarti hasil pencarian struktur tetangga selalu mendapatkan nilai tersebut setelah menentukan jumlah iterasi dan faktor penurun suhu tertentu.

5.2.3.3 Faktor Penurun Suhu dan Jumlah Iterasi Fase Improvement

Hasil dari *initial solution* masih terjebak dalam lokal optimal menyebabkan diperlukannya pencarian solusi tetangga yang memungkinkan memberikan hasil yang lebih baik. Pencarian solusi struktur tetangga dilakukan dengan menggunakan VNS dimana metode VNS yang

digunakan mirip dengan konsep dari *Simulated Annealing*. Maka dari itu diperlukannya input berupa Tawal yang mana merupakan *makespan* awal, alpha sebagai faktor penurun suhu dan jumlah iterasi untuk setiap keadaan suhu tertentu. Dalam melakukan proses *running* sebelum-sebelumnya alpha yang digunakan adalah 0.2 dengan maksimum iterasi sebanyak 200. Dalam analisa sensitivitas ini akan dilihat bagaimana pengaruh perubahan jumlah iterasi dan faktor penurun suhu terhadap hasil yang diinginkan yaitu *total setup time* dan *makespan*.

Berdasarkan hasil percobaan dapat dilihat bahwa faktor penurun suhu dan maksimum iterasi pada lini produksi *powder* tidak memberikan dampak perbedaan terhadap solusi hasil algoritma. VNS dengan faktor suhu dan maksimum iterasi dengan nilai berapapun mampu memperbaiki *initial solution setup time* menjadi 10.6 dan *makespan* menjadi 322.28. Sedangkan faktor penurun suhu dan maksimum iterasi pada produk *liquid* memberikan hasil yang berbeda-beda dalam upaya memperbaiki *initial solution*. Hal ini dapat terjadi karena variasi produk *powder* yang diproduksi pada bulan April tidak banyak sehingga memudahkan dalam melakukan pencarian terhadap seluruh struktur tetangga. Ketika jumlah variasi semakin sedikit maka kombinasi terhadap kemungkinan urutan juga semakin terbatas. Hal inilah yang kemudian menyebabkan pada produk *powder* dapat mencapai hasil yang sama untuk *setup time* dan *makespan* (mencapai *global optimum*). Sedangkan pada produk *liquid* variasinya sangat tinggi yaitu lebih dari 30 variasi produk. Ketika variasi semakin banyak maka kombinasi urutan dan peletakan produk dalam lini juga semakin beragam sehingga pencarian struktur tetangga dipengaruhi oleh maksimum iterasi dan faktor penurun suhu (alpha). Dalam Tabel 5.66 dapat dilihat bagaimana nilai faktor penurun suhu dan maksimum iterasi yang berbeda menyebabkan *total setup time* dan *makespan* juga berbeda. Faktor penurun suhu yang terlalu besar dapat menyebabkan proses pencarian struktur tetangga cepat konvergen dengan kata lain terlalu cepat menuju ke satu titik yang mana bukan *global optimum*. Didapatkan hasil yang terbaik adalah maksimum iterasi 200 dan faktor penurun suhu 0.6. Namun kedua hal ini tidak menjamin akan

memberikan hasil yang sama apabila dilakukan percobaan kembali. Maka dari itu penentuan maksimum iterasi dan faktor penurunan suhu harus tepat agar dapat memberikan *improvement* terbaik bagi *initial solution*.

(Halaman ini sengaja dikosongkan)

BAB 6

KESIMPULAN DAN SARAN

Pada bab ini akan dijelaskan mengenai kesimpulan dari hasil analisis dan pembahasan yang sesuai dengan tujuan penelitian. Selain itu juga akan diberikan saran untuk penelitian berikutnya yang memiliki permasalahan yang sama dalam penjadwalan.

6.1 Kesimpulan

Berdasarkan hasil uji coba dan analisis yang telah dilakukan pada bab sebelumnya, berikut adalah kesimpulan yang dapat diambil dari dilakukannya penelitian ini:

1. Algoritma heuristik untuk lini paralel identik dan *uniform* telah berhasil dikembangkan. Algoritma yang dikembangkan dibagi menjadi dua fase yaitu fase *constructive* berupa LFJ-LPT dan *Hungarian Method* serta fase *improvement* berupa *Variable Neighborhood Search* (VNS). Fase *improvement* dilakukan setelah mempertimbangkan bahwa fase *constructive* tidak begitu memuaskan sehingga VNS digunakan untuk memperbaiki hasil fase *constructive*. Algoritma yang terpilih ini mampu mengakomodasi pertimbangan-pertimbangan yang ada yaitu *sequence dependent setup time*, *machine eligibility restriction* dan *release time*.
2. Faktor penurunan suhu dan jumlah iterasi pada fase *improvement* memberikan pengaruh terhadap hasil algoritma sehingga penentuan kedua hal ini harus tepat agar memberikan hasil yang memperbaiki fase *constructive* dengan baik.
3. Fase *constructive* pada penjadwalan lini paralel *uniform* dengan menggunakan *Hungarian Method* belum menjamin minimasi *makespan* dan keseimbangan beban antar lini. Selain itu, kekurangan dari pengembangan algoritma ini adalah tidak dilakukannya penjadwalan urutan baru terhadap produk yang belum diproduksi karena datangnya produk MTO.

4. Pengembangan algoritma yang telah dilakukan berhasil mengurangi *total setup time* pada penjadwalan produksi produk *powder* sebesar 9.47% dan pada produk *liquid* sebesar 17.56%. Hal ini berarti *duration loss* terkait *setup time* berhasil berkurang sehingga nilai availabilitas dapat meningkat.
5. Berkurangnya *overtime* sebesar 35.29% menunjukkan algoritma yang dikembangkan mampu menjawab tujuan minimasi *makespan* pada penjadwalan operasi produksi produk *powder* dan *liquid*.
6. Model penjadwalan sistem aktual belum mempertimbangkan *sequence dependent setup time* sehingga hal tersebut menyebabkan total waktu *setup* menjadi tinggi. Pada algoritma heuristik yang dikembangkan telah mempertimbangkan *sequence dependent setup time* sehingga total *setup time* dapat diminimasi. Selain itu juga pada algoritma telah menerapkan LFJ-LPT yang bertujuan untuk meminimasi *makespan* sehingga dapat meminimasi kemungkinan terjadinya *overtime*.
7. Untuk lini yang bersifat *sharing*, alokasikan seluruh produk ke salah satu lini *sharing* pada awal bulan. Hal ini bertujuan agar kedatangan MTO pada bulan yang termasuk *peak season* dapat langsung diproduksi pada lini yang tidak digunakan. Harapannya dengan dilakukan hal ini dapat meminimasi terjadinya *overtime*. Sedangkan penjadwalan saat *normal season* dapat teratasi dengan baik dalam menjawab minimasi *total setup* dan *makespan* dengan algoritma ini.

6.2 Saran

Saran yang dapat dipertimbangkan untuk melakukan penelitian selanjutnya adalah sebagai berikut:

1. Perpaduan heuristik lain ataupun pendekatan metaheuristik dapat digunakan dalam rangka meningkatkan hasil penjadwalan yang lebih baik sehingga tidak diperlukannya fase *improvement*.
2. Mengembangkan model matematis yang mempertimbangkan *sequence dependent setup time*, *release time* dan *machine eligibility restriction* pada mesin paralel yang bersifat *uniform* dan *identic*.

DAFTAR PUSTAKA

- Baker, K. R., 1974. *Introduction to Sequencing and Scheduling*. New York: Jhon Willey and Sons, Inc.
- Baker, K. R. & Trietsch, D., 2009. *Principle of Sequencing and Scheduling*. New Jersey: John Wiley & Sons, Inc..
- Centeno, G. & Armacost, R. L., 2013. Minimizing Makespan on Parallel Machines with Release Time and Machine Eligibility Restriction. *International Journal of Production Research*, pp. 1243-1256.
- Chandran, K. S., 2009. *Modern Approach to Overall Equipment Effectiveness (OEE)*, s.l.: s.n.
- Chuang, M. C., Liao, C. J. & Chao, C. W., 2010. Parallel Machine Scheduling with Preference of Machines. *International Journal of Production Research*, Volume 48, pp. 4139-4152.
- Fogarty, D. W., Blackstone, J. H. & Hoffman, T. R., 1991. *Production and Inventory Management*. 2nd ed. Cincinnati: South Western Publishing Co..
- Gupta, J. N. D. & Chantaravarapan, S., 2007. Single Machine Group Scheduling with Family Setups to Minimize Total Tardiness. *International Journal of Production Research*, Volume 46, pp. 1707-1722.
- Huang, R. H. & Yu, T. H., 2010. *An Effective Heuristic Considering Machine Flexibility for Parallel Machine with Eligibility Problem*. Taipei, IEEE.
- Kuo, Y., Chen, S.-I. & Yeh, Y. H., 2017. Single Machine Scheduling with Sequence Dependent Setup Times and Delayed Precedence Constraints. *Journal International of Operational Research*, pp. 1-16.
- Pinedo, M. L., 2008. *Scheduling Theory, Algorithm and System*. 3rd ed. New York: Prentice Hall.
- Septiwianto, A., 2017. *Penjadwalan Multi Produk pada Mesin Paralel Filling Lithos di PT Pertamina Lubricants Plant Unit Gresik*. Surabaya: Departemen Teknik Industri FTI-ITS.
- Vazirani, V. V., 2003. *Approximation Algorithm*. Atlanta: Springer-Verlag Berlin Heidelberg.

(Halaman ini sengaja dikosongkan)

BIOGRAFI PENULIS



Penulis bernama Monica Pratiwi, lahir di Solo pada tanggal 27 Agustus 1997. Penulis merupakan anak ketiga dari empat bersaudara dari pasangan Ezra Simios Soetjipto dan Maria Padmasih Faktarina. Penulis menempuh Pendidikan di SDK St. Lusia Bekasi (2002-2009), SMPK St. Yosef Surabaya (2009-2012), SMAK St. Louis 1 Surabaya (2012-2015) dan Teknik Industri ITS (2015-2018).

Selama perkuliahan, penulis tercatat aktif mengikuti organisasi kemahasiswaan sebagai staf Departemen Sosial Masyarakat HMTI ITS (2016/2017), sekretaris “Dance Competition” ITS EXPO 2017 maupun kepanitiaan acara ormawa internal maupun eksternal seperti Mentor GERIGI ITS, Panitia SISTEM 2016, dan lain-lain. Semenjak menduduki semester 5, penulis tercatat sebagai Asisten *Quantitative Modelling and Industrial Policy Analysis Laboratory* (QMIPA Laboratory) serta pada kepengurusan QMIPA 2018/2019 diberikan kepercayaan menjadi *trainer* pada pelatihan *software* Lingo 2018. Selama menjadi asisten, penulis diberi kepercayaan sebagai sekretaris pada kepengurusan QMIPA 2017/2018. Saat menduduki tahun ketiga, penulis melakukan kerja praktek di PT Bayer Indonesia divisi *Cropscience* yang mana merupakan objek dari tugas akhir ini. Selain itu penulis juga mendapatkan beasiswa kepemimpinan selama dua tahun dari PT XL Axiata Tbk melalui XL Future Leaders batch 6. Penulis dapat dihubungi melalui email monicapratiwi1997@gmail.com. Terima kasih.